



آموزش

HTML

www.tahlildadeh.com

نویسنده: مهندس افшин رفوآ

برای دریافت سورس کد، پروژه های داخل کتاب به سایت مراجعه فرمایید.

1

آدرس آموزشگاه: تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک - 165 واحد 7
88146323 - 88446780 - 88146330





تقدیم به همه جویندگان علم که توان و امکان شرکت در کلاس های حضوری مارازم ازند.

فهرست

10	آموزش مقدمات HTML
10	پیش نیاز ها
11	مروری بر HTML
11	چیست؟ HTML
11	دکیومنت پایه ای HTML
12	برچسب های HTML
13	ساختار دکیومنت HTML
13	<!DOCTYPE>
14	برچسب های پایه ای در HTML
14	برچسب های تیتر
15	برچسب پاراگراف
15	برچسب شکست لینک
16	مرکزگذاری متن
17	خطوط افقی
18	حفظ طراحی
18	فضاهای غیرشکست
19	آموزش عناصر در HTML
20	برچسب HTML در مقابل عنصر
20	عناصر تو در توی HTML
21	ویژگی های زبان HTML
22	ویژگی های اصلی
23	ویژگی title
23	ویژگی class
24	ویژگی style
24	دروني کردن ویژگی ها
24	ویژگی dir
26	ویژگی xmlLang
26	ویژگی های عمومی
27	قالب بندی صفحات HTML
27	بولد کردن متن

28	ایتالیک کردن متن
28	آندر لاین کردن متن
29	خط کشیدن روی متن
29	فونت monospaced
30	متن ^(چاپ شده در بالا) superscript
30	متن _(چاپ شده در زیر) subscript
31	متن مندرج
31	متن حذف شده
31	متن بزرگتر
32	متن کوچکتر
32	گروه بندی محتوا
34	آموزش Meta Tag در HTML
34	افزودن برچسب های متأ به داکیومنت ها
35	مشخص کردن کلمات کلیدی
36	توصیف داکیومنت
36	بازبینی تاریخ داکیومنت
37	تازه سازی داکیومنت
37	کردن صفحه Redirect
38	تنظیم cookies
38	تنظیم نام نگارنده
39	تعیین تنظیم کاراکتر
40	آموزش استفاده از comment در html
40	کامنت های معتبر در مقابل کامنت های نامعتبر
41	کامنت های چندخطی
42	کامنت های شرطی
43	استفاده از برچسب کامنت
43	کد اسکریپت کامنت
44	کامنت صفحات طراحی
44	وارد کردن تصاویر در صفحات HTML
46	تنظیم موقعیت تصویر
47	تنظیم طول و عرض تصویر

48	تنظیم حاشیه‌ی تصویر
49	تنظیم همترازی تصویر
49	آموزش Table در HTML
50	تیتر جدول
50	ویژگی‌های cellspacing و cellpadding
51	ویژگی‌های rowspan و colspan
52	پس زمینه‌ی جدول
53	طول و عرض جدول
54	تیتر، بدن و پاورفی جدول
55	جدول‌های تودرتو
56	آموزش List
56	لیست‌های بدون ترتیب HTML
57	ویژگی type
58	لیست‌های منظم HTML
60	ویژگی start
61	لیست‌های تعریف HTML
61	لينک‌های متن HTML
62	لينک کردن داکیومنت‌ها
62	ویژگی target
64	استفاده از مسیر پایه
65	لينک شدن به بخشی از صفحه
65	تنظیم رنگ‌های لینک
66	HTML tahlildadeh
66	دانلود کردن لینک‌ها
66	Download PDF File
66	آموزش Image Link
67	تصاویر حساس به ماوس
69	سیستم مختصات
70	لينک ایمیل HTML
70	برچسب HTML Email
71	تنظیمات پیش‌فرض

71 آموزش HTML Layout
72 استفاده از جدول ها - HTML Layout
73 چند ستونی - استفاده از جدول ها Layout
74 HTML های - استفاده از DIV و Span Layout
75 آموزش Iframes
76 ویژگی های برچسب <iframe>
78 html background در
78 زمینه های HTML با رنگها
79 زمینه های HTML با تصاویر
80 زمینه های طرح دار و شفاف
81 رنگهای HTML
81 روش های کدگذاری رنگ HTML
82 رنگ های HTML - نام رنگ
82 16 رنگ استاندارد W3C
83 رنگ های HTML - کدهای Hex
84 رنگ های HTML - مقادیر RGB
88 فونت ها
89 تنظیم اندازه فونت
90 تنظیم ظاهر فونت
90 تعیین ظاهر فونت جایگزین
91 تنظیم رنگ فونت
91 عنصر <basefont>
92 فرم های html
92 ویژگی های فرم
93 کنترل های فرم HTML
94 کنترل های ورودی متن
94 کنترل های ورودی تک خطی
96 کنترل های پسورد ورودی
97 کنترل های متن ورودی چند خطی
98 کنترل چک باکس
99 کنترل دکمه رادیو

100	کنترل Select Box
102	فایل آپلود باکس
103	کنترل های دکمه
104	کنترل های مخفی شده ی فرم
104	آموزش چند رسانه ای در HTML
105	ویژگی های برچسب <embed>
106	انواع ویدیوهای پشتیبانی شده
107	صدا زمینه
109	آموزش HTML Marquee
112	سربرگ HTML
112	برچسب <title> در HTML
113	برچسب <meta> در HTML
114	برچسب <base> در HTML
114	برچسب <link> در HTML
115	برچسب <style> در HTML
116	برچسب <script> در HTML
116	آموزش Cascading style sheet
118	Style sheet خارجی
119	Style sheet داخلی
120	Inline Style Sheet
121	آموزش javascript
121	جاوا اسکریپت خارجی
122	جاوا اسکریپت درونی
123	گردانندگان رویدادها (EventHandler)
123	مخفي کردن اسکریپت ها از مرورگرهای قديمي
124	عنصر <noscript>

مقدمه

زکات علم نشر آن است. حرب علیم

موسسه آموزشی تحلیل داده ، با حضور جمعی از متخصصین مجبوب در زمینه برنامه نویسی در نظر دارد، مطالب آموزشی خود را در قالب کتاب های آموزشی و فیلم ، به صورت رایگان در دسترس عموم قرار دهد تا حتی آن دسته از عزیزانی که بنا به دلایل مالی، مسافت جغرافیایی و یا نداشتن وقت کافی ، امکان شرکت در دوره های حضوری برای آنها میسر نیست، از یادگیری بی بهره نمانند.

علاوه بر این علاقه مندان می توانند ، با ثبت نام در انجمن سایت تحلیل داده، سوالات خود را مطرح نموده و مدرسین آموزشگاه و اعضای انجمن در اسرع وقت، پاسخ های خود را، حتی الامکان به صورت فیلم، در دسترس عموم قرار دهند.

لذا از کلیه فعالان در این زمینه دعوت می شود، در این حرکت جمعی در کنار ما باشند و با حضور فعال خود در انجمن، گام موثری در بهبود سطح علمی جوانان کشور عزیzman، ایران بردارند.

ترجمه:

برای دانلود سورس کد مثال های کتاب، به بخش مقالات **Html** در آدرس www.tahlildadeh.com مراجعه فرمایید.

آموزش مقدمات HTML

HTML مخفف عبارت Hyper Text Markup Language می باشد که در ظاهر گسترش دهنده ترین زبان استفاده شده در صفحات وب می باشد.

HTML توسط برنز لی در اواخر سال 1991 ایجاد شد، اما html 2.0 اولین نوع استاندارد آن بود که در سال 1995 منتشر شد. HTML 4.01 ورژن مهمی از HTML بود که در اوایل 1999 منتشر شد. گرچه HTML 4.01 ورژنی است که به طور گسترش دهنده استفاده می شود، اما معمولاً HTML 5 را داریم که توسعه یافته است و در سال 2012 منتشر شد.

این آموزش برای آشنایی کردن طراحان و توسعه دهندگان وب با نیاز برای درک HTML با جزئیات کافی به همراه یک نظر کلی ساده و مثال های عملی، طراحی شده است. این آموزش به شما محتوای کافی برای شروع با HTML از جایی که می توانید تخصصی با سطح بالاتر داشته باشید، ارائه می دهد.

پیش نیازها

قبل از پیش رفتمن با این آموزش شما باید یک دانش پایه برای کار با ویندوز و سیستم عملیاتی Unix داشته باشید. علاوه بر این شما باید با موارد زیر آشنایی باشید

تجربه کار با هر ویرایشگری مانند Notepad ++ و Notepad و Editplus و غیره. چگونگی ایجاد مسیرها و فایل ها روی کامپیوتر خود.

چگونگی مسیریابی از طریق مسیرهای مختلف.

چگونگی تایپ کردن محتوا در یک فایل و ذخیره ی آنها روی کامپیوتر.

درک تصاویر با فرمات های مختلف مانند JPEG و PNG.

امتحان کردن HTML روی خط

مثال

```
<!DOCTYPE html>
<html>
<body>
<h1>Hello World!</h1>
```

```
</body>  
</html>
```

مروی بر HTML

Html چیست؟

می باشد که گسترده ترین زبان مورد استفاده برای نوشتن صفحات وب است.

اشاره به روشی دارد که در آن صفحات وب (دکیومنت های HTML) به هم لینک می شوند.

بنابراین لینک موجود روی یک صفحهٔ وب Hypertext نامیده می شود.
همانطور که از اسم آن پیداست HTML یک Markup language (زبان نشانه‌گذاری) می باشد، که بدین معناست که شما می توانید از HTML برای نشانه‌گذاری متن دکیومنت استفاده کنید با برچسب هایی که به مرورگر چگونگی ساخت آن را برای نمایش بیان می کند.

در واقع HTML به نیت تعریف ساختار دکیومنت هایی مانند تیترها، پاراگراف ها، لیست ها و غیره توسعه پیدا کرد تا اشتراک گذاری اطلاعات علمی بین محققان را آسان تر کند. اکنون HTML به طور گسترده‌ای مورد استفاده قرار می گیرد تا صفحات وب را به کمک برچسب های موجود در زبان HTML تنظیم کند.

دکیومنت پایهٔ HTML

به ساده ترین شکل خود، در زیر مثالی را از یک دکیومنت HTML مشاهده می کنید

```
<!DOCTYPE html>  
<html>  
<head>  
<title>This is document title</title>  
</head>  
<body>  
<h1>This is a heading</h1>  
<p>Document content goes here.....</p>  
</body>  
</html>
```

برای چک کردن نتیجه این کد HTML هم می توانید با استفاده از ویرایشگر مورد علاقهٔ خود، آن را در یک فایل HTML و با نام test.htm ذخیره کنید. در انتها آن را با استفاده از یک مرورگر مانند اینترنت اکسپلورر یا گوگل کروم یا فایرفاکس و غیره باز کنید.

برچسب های HTML

همانطور که قبل اگفته شد HTML یک زبان نشانه گذاری است و برای طراحی محتوا از برچسب های مختلفی استفاده می کند. این برچسب ها در داخل علامت `<>` قرار گرفته اند، به این شکل `<نام برچسب>`.

به جز تعداد کمی از برچسب ها، بسیاری از آنها برچسب متناظر بستن را نیز دارند. به عنوان مثال `<html>` دارای برچسب بستن `</html>` و `<body>` دارای برچسب بستن `</body>` می باشد.

مثال بالا از داکیومنت HTML از برچسب های زیر استفاده می کند

Tag	Description
<code><!DOCTYPE...></code>	این برچسب نوع داکیومنت و ورژن HTML را تعریف می کند.
<code><html></code>	این برچسب داکیومنت کامل HTML را احاطه می کند و اساسا شامل تیتر داکیومنت می شود که توسط <code><head>...</head></code> نمایش داده می شود و بدنه ی داکیومنت نیز به وسیله ی برچسب های <code><body>...</body></code> نمایش داده می شوند.
<code><head></code>	این برچسب تیتر داکیومنتی را نشان می دهد که می تواند دیگر برچسب های HTML از مانند <code><title></code> , <code><link></code> را در خود نگه دارد.
<code><title></code>	برچسب <code><title></code> در داخل برچسب <code><head></code> استفاده می شود تا تیتر داکیومنت را ذکر کند.
<code><body></code>	این برچسب بدنه ی داکیومنتی را نشان می دهد که دیگر برچسب های HTML مانند <code><h1></code> , <code><div></code> , <code><p></code> را در خود دارد.
<code><h1></code>	این برچسب تیتر را نمایش می دهد.

<p>	این برچسب یک پاراگراف را نمایش می دهد.
-----	--

برای یادگیری **HTML** لازم است برچسب های مختلفی را مطالعه کنید و درک کنید که در هنگام طراحی یک داکیومنت متنی چگونه رفتار می کنند. یادگیری **HTML** ساده است چرا که بیوزرها باید مورد استفاده ی برچسب های مختلف را برای طراحی متن و یا تصاویر و ایجاد یک صفحه وب زیباتر یاد بگیرند.

وب جهانی **(W3C Consortium)** پیشنهاد می دهد که در شروع با **HTML** برچسب ها را با حروف کوچک به کار ببرید.

Document declaration tag

```
<html>
<head>
  Document header related tags
</head>
<body>
  Document body related tags
</body>
</html>
```

یک نمونه از داکیومنت **HTML** دارای ساختار زیر خواهد بود

ساختار داکیومنت **HTML**

تمام برچسب های مربوط به بدنه **HTML** را در بخش های بعدی مطالعه خواهیم کرد، اکنون اجازه بدھید در مورد برچسب اطلاعیه ی داکیومنت بیشتر بدانیم.

برچسب <!DOCTYPE>

برچسب اطلاعیه ی **<!DOCTYPE>** توسط مرورگرهای وب برای پی بردن به ورژن **HTML** استفاده شده در داکیومنت مورد استفاده قرار می گیرد. ورژن حال حاضر **HTML 5** می باشد که از اطلاعیه ی زیر استفاده می کند

```
<!DOCTYPE html>
```

انواع دیگری اطلاعیه وجود دارند که می توانند در داکیومنت **HTML**، متناسب با ورژن **HTML** مورد استفاده قرار بگیرند. جزئیات بیشتری در این مورد در زمان بحث در مورد برچسب های دیگر **HTML** فرا خواهیم گرفت.

برچسب های پایه ای در **HTML**

برچسب های تیتر

هر داکیومنت با یک تیتر آغاز می شود. شما می توانید از اندازه های مختلف برای تیترهای خود استفاده کنید. **HTML** دارای شش سطح می باشد که از **<h1>**, **<h2>**, **<h3>**, **<h4>**, **<h5>** و **<h6>** عناصر استفاده می کند. در هنگام نمایش هر تیتر مرور گر یک خط قبل و یک خط بعد از تیتر اضافه می کند.

```
<!DOCTYPE html>
<html>
<head>
  <title>Heading Example </title>
</head>
<body>
  <h1>This is heading 1 </h1>
  <h2>This is heading 2 </h2>
  <h3>This is heading 3 </h3>
  <h4>This is heading 4 </h4>
  <h5>This is heading 5 </h5>
  <h6>This is heading 6 </h6>
</body>
</html>
```

مثال

آموزشگاه حلیلکرداده

این مثال نتیجه زیر را به دنبال دارد

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

برچسب پاراگراف

برچسب `<p>` به روش طراحی متن شما در پاراگراف های مختلف اشاره دارد. هر پاراگراف متن باید بین برچسب بازگشته `<p>` و بستن `</p>` قرار بگیرد، همانطور که در مثال زیر مشاهده می کنید

```
<!DOCTYPE html>
<html>
<head>
  <title>Paragraph Example </title>
</head>
<body>
  <p>Here is a first paragraph of text. </p>
  <p>Here is a second paragraph of text. </p>
  <p>Here is a third paragraph of text. </p>
</body>
</html>
```

برچسب شکست لینک

هر زمان که شما از `
` عنصر استفاده کنید، هر چیزی که آن را دنبال می کند از خط بعد شروع خواهد شد.
این برچسب نمونه ای از یک عنصر `empty` می باشد، زمانی که لازم نیست برچسبی را باز کنید یا ببندید چرا که چیزی برای رفتن بین آنها وجود ندارد.

در بچسب `
`، یک فضای خالی بین کاکترهای `br` و اسلش جلوی آن وجود دارد. اگر شما این فضا را حذف کنید، مرورگرهای قدیمی تر در اجرای خط شکست مشکل خواهند داشت، در حالیکه اگر اسلش را حذف کنید برچسب `
` باقیمانده در `HTML` معتبر نمی باشد.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Line Break Example </title>
</head>
<body>
  <p>
    Hello <br/>
    You delivered your assignment ontime. <br/>
    Thanks <br/>
    Mahnaz
  </p>
</body>
</html>
```

مثال بالا نتیجه‌ی زیر را تولید خواهد کرد

Hello
You delivered your assignment ontime.
Thanks
Mahnaz

مرکزگذاری متن

می‌توانید با استفاده از برچسب `<center>` می‌توانید هر محتوایی را در مرکز صفحه یا در مرکز هر سلول از یک جدول قرار دهید.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Centring Content Example </title>
</head>
<body>
  <p>This text is not in the center. </p>
  <center>
    <p>This text is in the center. </p>
  </center>
</body>
</html>>
```

مثال بالا نتیجه‌ی زیر را تولید خواهد کرد

This text is not in the center.

This text is in the center.

خطوط افقی

خطوط افقی برای بخش‌های شکست بصری یک داکیومنت استفاده می‌شوند. برچسب `<hr>` خطی از موقعیت کنونی داکیومنت به حاشیه‌ی سمت راست ایجاد کرده و خط را طبق آن می‌شکند.

به عنوان مثال ممکن است تمایل داشته باشید بین دو پاراگراف خطی قرار دهید، همانطور که در مثال زیر ارائه شده است

```
<!DOCTYPE html>
<html>
<head>
  <title>Horizontal Line Example </title>
</head>
<body>
  <p>This is paragraph one and should be on top </p>
  <hr />
  <p>This is paragraph two and should be at bottom </p>
</body>
</html>
```



مثال

این مثال نتیجه‌ی زیر را تولید خواهد کرد

This is paragraph one and should be on top

آموزشگاه حلیلکرد

This is paragraph two and should be at bottom

و باز برچسب `<hr />` مثالی از `Empty` می‌باشد که نیازی به باز کردن یا بستن برچسب ندارید زیرا چیزی برای رفتن بین آنها وجود ندارد.

در عنصر `<hr>` یک فضای خالی بین کاراکترهای `hr` و اسلش مقابله آن وجود دارد. اگر این فضا را حذف کنید مروارگرهای قدیمی تر در اجرای خط افقی مشکل خواهند داشت. در حالیکه اگر اسلش مقابله آن را حذف کنید عنصر باقیمانده `<hr>` می باشد که در `HTML` فاقد اعتبار می باشد.

حفظ طراحی

گاهی اوقات تمایل دارید که متن فرمت دقیق خود در `HTML` را دنبال کند، در این موارد می توانید از برچسب پریفرمت `<pre>` استفاده کنید.

هر متن بین برچسب باز کننده `<pre>` و برچسب بستن `</pre>` طراحی متن منبع را حفظ خواهد کرد.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Preserve Formatting Example </title>
</head>
<body>
<pre>
function testFunction( strText ){
  alert (strText)
}
</pre>
</body>
</html>
```



آموزشگاه تحلیلکرداری

مثال بالا نتیجه‌ی زیر را تولید خواهد کرد

```
function testFunction( strText ){
  alert (strText)
}
```

سعی کنید از همان کد بدون نگهداشتن آن بین برچسب های `<pre>...</pre>` استفاده کنید.

فضاهای غیرشکست

فرض کنید می خواهید از عبارت "12 Angry Men." استفاده کنید. در اینجا از مروارگر نمی خواهید عبارت را بین دو خط به صورت `Men` و `12 Angry` بشکند.

An example of [this](#) technique appears in the movie "12 Angry Men."

در مواردی که نمی خواهید مرورگر متن را بشکند باید به جای یک فضای عادی از فضای غیر شکست استفاده کنید. برای مثال وقتی "12 Angry Men" را در یک پاراگراف کدگذاری می کنید باید از کدی مانند زیر استفاده کنید

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Nonbreaking Spaces Example</title>
</head>
<body>
  <p>An example of this technique appears in the movie "12 &nbsp;Angry&nbsp;Men."</p>
</body>
</html>
```

آموزش عناصر در HTML

یک عنصر HTML توسط یک برچسب شروع کننده تعریف می شود. اگر عنصر دارای محتوای دیگری باشد، با یک برچسب بسته کننده تمام می شود در حالیکه جلوی نام عنصر یک علامت اسلش قرار گرفته است که می توانید در جدول زیر برخی از این برچسب ها را مشاهده کنید

Start Tag	Content	End Tag
<p>	این برچسب محتوای پاراگراف می باشد.	</p>
<h1>	این برچسب تیتر محتوا می باشد.	</h1>
<div>	این برچسب تقسیم محتوا می باشد.	</div>

بنابراین در اینجا `<p>....</p>` یک عنصر HTML و `<h1>....</h1>` عنصر دیگری از HTML می باشد. عناصری از HTML وجود دارند که نیازی به بسته شدن ندارند مانند `
`, `<hr />` و `<img.../>`. این عناصر با عنوان void elements (عناصر خالی) شناخته می شوند.

دکیومت های HTML دارای درختی از این عناصر می باشند و مشخص می کنند که چگونه دکیومت ها باید ساخته شوند و چه نوع محتوایی باید در چه بخشی از دکیومت HTML قرار بگیرد.

برچسب HTML در مقابل عنصر

یک عنصر HTML به وسیله ی یک برچسب شروع کننده تعریف می شود. اگر عنصر دارای محتوای دیگری باشد با یک برچسب بستن تمام می شود.

برای مثال `<p>` برچسب شروع کننده ی یک پاراگراف می باشد و `</p>` برچسب بستن همان پاراگراف می باشد، اما `<p>This is paragraph</p>` عنصر یک پاراگراف می باشد.

عناصر تو در توی HTML

نگذاری یک عنصر HTML در داخل عنصر دیگر بسیار متداول می باشد.

```
<!DOCTYPE html>
<html>
<head>
  <title>Nested Elements Example </title>
</head>
<body>
  <h1>This is <i>italic</i> heading</h1>
  <p>This is <u>underlined</u> paragraph</p>
</body>
</html>
```

مثال بالا نتیجه ی زیر را دنبال خواهد کرد

This is *italic*heading

This is underlined paragraph

ویژگی های زبان HTML

برخی از برچسب های HTML مانند برچسب های تیتر و برچسب های پاراگراف، و موارد استفاده‌ی آنها را مشاهده کردیم. تاکنون از آنها به ساده ترین شکل خود استفاده کرده ایم، اما بیشتر برچسب های HTML می‌توانند ویژگی هایی داشته باشند که مقداری اطلاعات اضافه می‌باشد.

یک attribute برای تعریف ویژگی های عنصر HTML استفاده می‌شود و در داخل برچسب بازگشته‌ی عنصر قرار می‌گیرد. همه‌ی ویژگی ها از دو بخش تشکیل شده‌اند name و value.

ویژگی مورد نظر شما برای تنظیم می‌باشد، به عنوان مثال عنصر پاراگراف <p> در مثال ارائه شده دارای ویژگی می‌باشد که نام آن align می‌باشد و شما می‌توانید از آن برای تنظیم پاراگراف در صفحه استفاده کنید.

همان است که شما می‌خواهید مقدار ویژگی تنظیم شود و همیشه در داخل گیوه قرار می‌دهید. مثال زیر سه مقدار ممکن از یک ویژگی تراز را نشان می‌دهد چپ، مرکز و راست.

ویژگی نام‌ها و ویژگی مقادیر غیرهشمند می‌باشند. به هر حال وب جهانی W3C Consortium مقادیر ویژگی‌ها را در HTML 4 با حروف کوچک پیشنهاد می‌دهد.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Align Attribute Example</title>
</head>
<body>
  <p align="left">This is left aligned</p>
  <p align="center">This is center aligned</p>
  <p align="right">This is right aligned</p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را دنبال خواهد کرد

This is left aligned

This is center aligned

This is right aligned

ویژگی های اصلی

چهار ویژگی اصلی که می توانند در اکثر عناصر HTML مورد استفاده قرار بگیرند عبارتند از

id

title

class

style

ویژگی id یک برچسب HTML می تواند برای تشخیص یک عنصر در یک صفحه HTML مورد استفاده قرار بگیرد. دو دلیل اصلی برای تمایل شما به استفاده از ویژگی id در یک عنصر وجود دارد

اگر یک عنصر یک ویژگی را به عنوان تشخیص دهنده منحصر به فرد استفاده می کند شناخت تنها آن عنصر و محتوای مربوط به آن ممکن می باشد.

اگر دارای دو عنصر هم نام در یک صفحه وب می باشد، می توانید از عنصر id برای تشخیص این عناصر هم نام استفاده کنید.

در مورد طراحی صفحه در آموزش های مجزا بحث خواهیم کرد، اکنون اجازه بدھید از ویژگی id برای تشخیص عناصر بین دو پاراگراف استفاده کنیم، مانند مثال زیر

```
<p id="html">This para explains what is HTML</p>
<p id="css">This para explains what is Cascading Style Sheet</p>
```

ویژگی title

این ویژگی یک تیتر پیشنهادی برای عنصر ارائه می‌دهد. ترکیب مربوط به ویژگی **title** شبیه به ترکیب توضیح داده شده برای ویژگی **id** می‌باشد. رفتار این ویژگی بستگی به عنصری دارد که آن را حمل می‌کند، گرچه اغلب اوقات وقتی مکان نما روی عنصر قرار می‌گیرد یا عنصر در حال بارگذاری می‌باشد، با عنوان یک راهنمای ابزار (**tooltip**) نمایش داده می‌شود.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>The title Attribute Example</title>
</head>
<body>
  <h3 title="Hello HTML!">Titled Heading Tag Example</h3>
</body>
</html>
```

مثال بالا نتیجه‌ی زیر را تولید خواهد کرد

Titled Heading Tag Example

اکنون سعی کنید مکان نما را روی "Titled Heading Tag Example" بیاورید، خواهید دید که هر تیتری که در کد خود استفاده کرده‌اید، مانند یک راهنمای ابزار نمایش داده خواهد شد.

ویژگی class

این ویژگی برای برقراری ارتباط بین یک عنصر با یک صفحه‌ی طراحی استفاده می‌شود و گروه عنصر را مشخص می‌کند. وقتی که را یاد بگیرید **Cascading Style Sheet (CSS)** در مورد این ویژگی بیشتر فراخواهید گرفت. اکنون تا همین حد کافیست.

مقدار ویژگی نیز ممکن است لیستی از فضاهای مجزای نام‌های گروه باشد، برای مثال

```
class="className1 className2 className3"
```

ویژگی style

این ویژگی به شما اجازه می دهد تا قوانین CSS را در داخل عنصر مشخص کنید.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>The style Attribute</title>
</head>
<body>
  <p style="font-family:arial; color:#FF0000;">Some text...</p>
</body>
</html>
```

Some text...

این مثال نتیجه‌ی زیر را تولید می‌کند

اکنون CSS را یاد نمی‌گیریم بنابراین اجازه دهید بدون اینکه بیشتر از این خود را در مورد CSS اذیت کنیم، به جلوتر برویم. در اینجا لازم است در مورد ویژگی‌های HTML بدانید و اینکه چگونه در هنگام طراحی متن مورد استفاده قرار می‌گیرند.

درونوی کردن ویژگی‌ها

سه ویژگی درونی وجود دارند که برای اکثر عناصر HTML استفاده می‌شوند

dir

lang

xmllang

ویژگی dir

این ویژگی به شما اجازه می‌دهد تا مسیری را به مرورگر نشان دهید که متن در آن باید جریان داشته باشد.

ویژگی dir می‌تواند یکی از دو مقدار باشد، همانطور که در جدول زیر نشان داده شده است

Value	Meaning
ltr	از چپ به راست (مقدار پیش فرض)
rtl	راست به چپ (برای زبان هایی مانند هبرو یا عربی که از راست به چپ خوانده می شوند.)

```

<!DOCTYPE html>
<html dir="rtl">
<head>
  <title>Display Directions</title>
</head>
<body>
  This is how IE 5 renders right-to-left directed text.
</body>
</html>

```

مثال

این مثال نتیجه‌ی زیر را تولید خواهد کرد

This is how IE 5 renders right-to-left directed text.

وقتی که ویژگی `dir` در داخل برچسب `<html>` استفاده می‌شود، مشخص می‌کند که چگونه متن در کل داکیومنت نمایش داده می‌شود. وقتی در برچسب دیگری مورد استفاده قرار بگیرد، مسیر متن را برای محتوای مربوط به آن برچسب کنترل می‌کند.

ویژگی `lang` این ویژگی به شما کمک می‌کند تا زبان اصلی استفاده شده در یک داکیومنت را نشان دهید، اما این ویژگی فقط برای سازگاری معکوس با ورژن‌های قدیمی تر `HTML` در `HTML` حفظ شده است. این ویژگی به وسیله‌ی ویژگی `xmllang` در داکیومنت‌های جدید `HTML` جایگزین شده است.

مقادیر ویژگی `lang` کدهای دو کاراکتری زبان استاندارد `ISO-639` می‌باشند. `HTML Language Codes` را برای لیست کاملی از کدهای زبان چک کنید.

مثال

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>English Language Page</title>
</head>
<body>
  This page is using English Language
</body>
</html>
```

xmlLang ویژگی

این ویژگی جایگزین **xhtml** برای **xmlLang** می باشد. مقدار **xmlLang** باید یک کد ISO-639 می باشد، همانطور که در بخش قبل بیان شد.

ویژگی های عمومی

در اینجا جدولی را می بینید که برخی دیگر از ویژگی هایی که در برچسب های **HTML** مفید می باشند، را نشان می دهد.

Attribute	Options	Function
align	راست، چپ، مرکز	برچسب ها را به طور افقی تنظیم می کند.
valign	بالا، وسط، پایین.	برچسب ها را به طور عمودی در یک عنصر HTML تنظیم می کند.
bgcolor	numeric, hexadecimal, RGB values	پشت یک عنصر یک رنگ پس زمینه قرار می دهد.
background	URL	پشت یک عنصر یک تصویر پس زمینه قرار می دهد.

id	User Defined	عنصری را برای استفاده با Cascading Style Sheets (CSS) نام می برد.
class	User Defined	عنصری را برای استفاده با Cascading Style Sheets طبقه بندی می کند.
width	Numeric Value	عرض جدول ها، تصاویر و یا سلول های جدول ها را مشخص می کند.
height	Numeric Value	طول جدول ها، تصاویر و یا سلول های جدول ها را مشخص می کند.
title	User Defined	تیتر عناصر را پاپ آپ می کند.

قالب بندی صفحات HTML

اگر با پردازشگر word کار می کنید باید با بولد کردن (**bold**)، ایتالیک کردن و آندرلاین کردن متن آشنا باشید. این ها فقط سه گزینه از ده گزینه موجود برای چگونگی ظاهر شدن متن در HTML و XHTML می باشند.

بولد کردن متن

...

قرار می گیرد به صورت بولد ظاهر می شود، مانند مثال زیر

مثال

```
<!DOCTYPE html>
<html>
<head>
```

```

<title>Bold Text Example</title>
</head>
<body>
  <p>The following word uses a <b>bold</b> typeface.</p>
</body>
</html>

```

مثال بالا نتیجه‌ی زیر را در برخواهد داشت

The following word uses a bold typeface.

```

<!DOCTYPE html>
<html>
<head>
  <title>Italic Text Example </title>
</head>
<body>
  <p>The following word uses a <i>italicized</i> typeface.</p>
</body>
</html>

```

ایتالیک کردن متن

هر چیزی که بین عناصر <i>...</i>

قرار بگیرد به صورت ایتالیک ظاهر می‌شود، مانند مثال زیر

مثال

The following word uses a *italicized* typeface.

آندرلاین کردن متن

هر چیزی که بین عناصر <u>...</u> قرار بگیرد به صورت آندرلاین ظاهر می‌شود، مانند مثال زیر

```

<!DOCTYPE html>
<html>
<head>
  <title>Underlined Text Example</title>
</head>
<body>
  <p>The following word uses a <u>underlined</u> typeface.</p>
</body>
</html>

```

مثال بالا نتیجه‌ی زیر را تولید خواهد کرد

The following word uses a underlined typeface.

خط کشیدن روی متن

هر چیزی که بین گزینه‌های `<strike>...</strike>` قرار بگیرد، با یک **strikethrough** نمایش داده می‌شود
که خط باریکی می‌باشد که روی متن کشیده می‌شود، همانطور که در مثال زیر مشاهده می‌کنید

```
<!DOCTYPE html>
<html>
<head>
  <title>Strike Text Example</title>
</head>
<body>
  <p>The following word uses a <strike>strikethrough</strike> typeface.</p>
</body>
</html>
```

مثال بالا نتیجه‌ی زیر را تولید خواهد کرد

The following word uses a ~~strikethrough~~ typeface.

فونت monospaced

محتوای عنصر `<tt>...</tt>` به فونت **monospaced** نوشته می‌شود. اکثر فونت‌ها با عنوان فونت‌هایی با عرض متغیر شناخته شده‌اند، زیرا حروف مختلف دارای عرض‌های مختلف هستند (به عنوان مثال حرف `m` عریض تر از حرف `a` می‌باشد). به هرحال در فونت **monospaced** تمام حروف دارای عرض یکسان می‌باشند.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Monospaced Font Example</title>
</head>
<body>
  <p>The following word uses a <tt>monospaced</tt> typeface.</p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید خواهد کرد

The following word uses a monospaced typeface.

متن ^(چاپ شده در بالا) superscript

محتوای عنصر ^{^{...}} در بالا نوشته می شود، فونت استفاده شده برای آن همان فونت کاراکترهاي اطراف آن می باشد، اما به اندازه ي نصف يك کاراکتر بالاتر از ديگر کاراکترها نمایش داده می شود.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Superscript Text Example</title>
</head>
<body>
  <p>The following word uses a <sup>superscript</sup> typeface.</p>
</body>
</html>
```

The following word uses a ^{superscript} typeface.

این مثال نتیجه ي زیر را تولید خواهد کرد

متن _(چاپ شده در زیر) subscript

محتوای عنصر _{_{...}} در زیر نوشته می شود. اندازه ي فونت استفاده شده برای آن به اندازه ي فونت کاراکترهاي اطراف می باشد اما به اندازه ي نصف ارتفاع يك کاراکتر زیر کاراکترهاي ديگر نمایش داده می شود.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Subscript Text Example</title>
</head>
<body>
  <p>The following word uses a <sub>subscript</sub> typeface.</p>
</body>
</html>
```

این مثال نتیجه ي زیر را تولید خواهد کرد

The following word uses a _{subscript} typeface.

متن مندرج

هرچیزی که بین عنصر **<ins>...</ins>** قرار بگیرد به عنوان متن مندرج نمایش داده می شود.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Inserted Text Example</title>
</head>
<body>
  <p>I want to drink <del>cola</del><ins>wine</ins></p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید خواهد کرد

I want to drink ~~cola~~ wine

متن حذف شده

هرچیزی که بین عنصر **...** ظاهر شود با عنوان یک متن حذف شده نمایش داده می شود.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Deleted Text Example</title>
</head>
<body>
  <p>I want to drink <del>cola</del><ins>wine</ins></p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید می کند

I want to drink ~~cola~~ wine

متن بزرگتر

محتوای عنصر **<big>...</big>** اندازه فونت را بزرگتر از متن اطراف نشان می دهد. مانند مثال زیر

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Larger Text Example</title>
</head>
<body>
  <p>The following word uses a <big>big</big> typeface.</p>
</body>
</html>
```

این مثال نتیجه زیر را تولید می کند

The following word uses a big typeface.

متن کوچکتر

محتوای عنصر ... متن را یک سایز کوچکتر از متن اطراف آن نشان می دهد، مانند مثال زیر

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Smaller Text Example</title>
</head>
<body>
  <p>The following word uses a <small>small</small> typeface.</p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید خواهد کرد

The following word uses a small typeface.

گروه بندی محتوا

عناصر و

به شما اجازه می دهند تا برای ایجاد بخش ها و یا زیرمجموعه های یک صفحه، عناصر زیادی را با یکدیگر گروه بندی کنید.

برای مثال ممکن است تمایل داشته باشید که تمام پاورقی ها را در یک صفحه در داخل عنصر `<div>` قرار دهید تا نشان دهید که تمام عناصر موجود در آن عنصر مربوط به پاورقی می باشند. پس از آن ممکن است طرحی را به عنصر `<div>` ضمیمه کنید، طوریکه با استفاده از مجموعه ای از قوانین طراحی ظاهر شوند.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Div Tag Example</title>
</head>
<body>
  <div id="menu" align="middle">
    <a href="/index.htm">HOME</a> | 
    <a href="/about/contact_us.htm">CONTACT</a> | 
    <a href="/about/index.htm">ABOUT</a>
  </div>
  <div id="content" align="left" bgcolor="white">
    <h5>Content Articles</h5>
    <p>Actual content goes here.....</p>
  </div>
</body>
</html>
```

HOME | CONTACT | ABOUT

CONTENT ARTICLES

Actual content goes here.....

این مثال نتیجه‌ی زیر را تولید خواهد کرد



از طرف دیگر عنصر `` فقط می تواند برای گروه بندی داخل خطی عناصر استفاده شود. بنابراین اگر بخشی از جمله یا پاراگراف را دارید که می خواهید با هم در یک گروه قرار دهید، می توانید از عنصر `` مانند زیر استفاده کنید

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Span Tag Example</title>
</head>
<body>
```

```
<p>This is the example of <span style="color:green">span tag</span> and the <span style="color:red">div tag</span> alongwith  
CSS</p>  
</body>  
</html>
```

این مثال نتیجه‌ی زیر را تولید خواهد کرد

This is the example of **span tag** and the **div tag** alongwith CSS

این برجسب‌ها عموماً با **CSS** استفاده می‌شوند تا به شما اجازه دهنده طرحی را به بخشی از یک صفحه ضمیمه کنند.

آموزش **Meta Tag** در **HTML**

علاوه بر مشخص کردن اطلاعات مهم به روش‌های مختلف در مورد یک داکیومنت، به شما اجازه می‌دهد تا متأذیتا را نیز مشخص کنید. عناصر **META** می‌توانند برای وارد کردن جفت مقدار/نام استفاده شوند، این جفت ویژگی‌های داکیومنت **HTML** مانند نویسنده، تاریخ اتمام، لیست کلمات کلیدی، داکیومنت نویسنده و غیره را ارائه میدهد.

برچسب **<meta>** برای ارائه‌ی چنین اطلاعات اضافه‌ای استفاده می‌شود. این برجسب یک عنصر خالیست و دارای برجسب بستن نیست اما اطلاعاتی را با ویژگی آن در خود دارد. شما می‌توانید بر اساس اطلاعاتی که می‌خواهید در داکیومنت خود نگهداری کنید، یک برجسب یا بیشتر از یک برجسب متن وارد داکیومنت خود کنید. اما به طور کل برجسب‌های متن وضعیت ظاهری داکیومنت را تحت تاثیر قرار نمی‌دهند، بنابراین از لحاظ ظاهری استفاده کردن یا نکردن از آنها مشخص نمی‌شود.

افزودن برجسب‌های متن به داکیومنت ها

شما می‌توانید با قرار دادن برجسب‌های **<meta>** در داخل تیتر داکیومنت که با برجسب‌های **<head>** و **</head>** مشخص می‌شوند، متن دیتا را به صفحات وب خود اضافه کنید. یک برجسب متن علاوه بر ویژگی‌های مرکزی، می‌تواند دارای ویژگی‌های زیر نیز باشد

Attribute	Description
Name	نام برای پردازشی، می تواند هر چیزی از جمله کلمات کلیدی، توصیفات، نویسنده، تولید کننده و غیره باشد.
content	مقدار ویژگی را مشخص می کند.
scheme	نموداری را برای توضیح مقدار ویژگی مشخص می کند. (همانطور که در محتوى ویژگی مشاهده کردید.)
http-equiv	برای تیترهای پیام پاسخ http استفاده می شود. برای مثال http-equiv می تواند برای تازه سازی صفحات یا تنظیم یک cookie استفاده شود. مقادیری مانند نوع محتوا، اتمام، تازه سازی و تنظیم cookie.

مشخص کردن کلمات کلیدی

می توانید از برچسب **<meta>** برای مشخص کردن کلمات کلیدی مربوط به داکیومنت استفاده کنید، و پس از آن این کلمات توسط موتورهای جستجو استفاده می شوند، و صفحه ی وب شما را به هدف جستجو ایندکس می کنند.

مثال

در این مثال برچسب های متأ، متادیتا و **HTML** را با عنوان کلمات کلیدی در مورد داکیومنت وارد می کنیم.

```
<!DOCTYPE html>
<html>
<head>
  <title>Meta Tags Example</title>
  <meta name="keywords" content="HTML, Meta Tags, Metadata" />
</head>
<body>
  <p>Hello HTML5!</p>
</body>
```

```
</html>
```

این مثال نتیجه‌ی زیر را به دنبال دارد

Hello HTML5!

توصیف داکیومنت

شما می‌توانید از برچسب **meta** برای توصیف داکیومنت استفاده کنید. این برچسب نیز می‌تواند توسط موتورهای مختلف جستجو مورد استفاده قرار بگیرد، در حالیکه صفحه‌ی وب شما را به هدف جستجو ایندکس می‌کند.

```
<!DOCTYPE html>
<html>
<head>
  <title>Meta Tags Example</title>
  <meta name="keywords" content="HTML, Meta Tags, Metadata" />
  <meta name="description" content="Learning about Meta Tags." />
</head>
<body>
  <p>Hello HTML5!</p>
</body>
</html>
```

مثال

بازبینی تاریخ داکیومنت

شما می‌توانید از برچسب **meta** برای ارائه‌ی اطلاعات در مورد زمان آخرین آپدیت داکیومنت استفاده کنید. این اطلاعات می‌توانند توسط مرورگرهای مختلفی استفاده شوند، در حالیکه صفحه‌ی وب شما را تازه‌سازی می‌کنند.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Meta Tags Example</title>
  <meta name="keywords" content="HTML, Meta Tags, Metadata" />
  <meta name="description" content="Learning about Meta Tags." />
  <meta name="revised" content="Tahliddeh, 3/7/2014" />
</head>
<body>
```

```
<p>Hello HTML5!</p>
</body>
</html>
```

تازه سازی داکیومت

یک برجسب **<meta>** می تواند برای مشخص کردن دوره ای که پس از آن صفحه ی وب شما به طور خودکار بازسازی می شود، استفاده شود.

مثال

اگر می خواهید صفحه ی وب شما پس از هر 5 ثانیه ریفرش شود، از ترکیب زیر استفاده کنید

```
<!DOCTYPE html>
<html>
<head>
  <title>Meta Tags Example</title>
  <meta name="keywords" content="HTML, Meta Tags, Metadata" />
  <meta name="description" content="Learning about Meta Tags." />
  <meta name="revised" content="Tahlidadeh, 3/7/2014" />
  <meta http-equiv="refresh" content="5" />
</head>
<body>
  <p>Hello HTML5!</p>
</body>
</html>
```

کردن صفحه Redirect

می توانید از برجسب **<meta>** برای Redirect کردن صفحه ی خود استفاده کنید، همچنین می توانید دوره ای را مشخص کنید که پس از آن صفحه به طور خودکار Redirect شود.

مثال

در این مثال صفحه ی جاری پس از 5 ثانیه به صفحه ی دیگر Redirect می شود. اگر می خواهید صفحه فوراً شود، هیچ محتوایی برای آن مشخص نکنید.

```
<!DOCTYPE html>
<html>
<head>
  <title>Meta Tags Example</title>
  <meta name="keywords" content="HTML, Meta Tags, Metadata" />
  <meta name="description" content="Learning about Meta Tags." />

```

```
<meta name="revised" content="Tahlildadeh, 3/7/2014" />
<meta http-equiv="refresh" content="5" url=http://www.tahlildadeh.com" />
</head>
<body>
  <p>Hello HTML5!</p>
</body>
</html>
```

cookies تنظیم

داده هایی هستند که در یک فایل کوچک متن روی کامپیوتر شما ذخیره شده اند و بین مرورگر وب و سرور وب ردو بدل می شود تا مسیر اطلاعات مختلف را براساس نیاز برنامه‌ی وب شما حفظ کنند.

شما می توانید از برچسب **cookies** برای ذخیره‌ی **cookies** در بخش کاربری استفاده کنید و پس از آن این اطلاعات می توانند توسط سرور وب استفاده شوند تا بازدیدکننده‌ی سایت را پیگیری کنند.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Meta Tags Example</title>
  <meta name="keywords" content="HTML, Meta Tags, Metadata" />
  <meta name="description" content="Learning about Meta Tags." />
  <meta name="revised" content="Tahlildadeh, 3/7/2014" />
  <meta http-equiv="cookie" content="userid=xyz; expires=Wednesday, 08-Aug-15 235959 GMT;" />
</head>
<body>
  <p>Hello HTML5!</p>
</body>
</html>
```

اگر تاریخ و زمان انقضا را مشخص نکرده‌اید، **session cookie** یک **cookie** می باشد و وقتی که بوزار از مرورگر خارج شود، پاک خواهد شد.

تنظیم نام نگارنده

می توانید با استفاده از **meta tag** نام یک نگارنده را برای صفحه‌ی وب خود تنظیم کنید. یک مثال در این رابطه را در زیر مشاهده می کنید.

مثال

```

<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="author" content="Mahnaz Mohtashim" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>

```

تعیین تنظیم کاراکتر

می توانید از برچسب **<meta>** برای تعیین تنظیم کاراکتر مربوط به صفحه‌ی وب استفاده کنید.

مثال

به طور پیش فرض مرورگرها و سرورهای وب از رمزگزاری **ISO-8859-1** برای پردازش صفحات وب استفاده می کنند. در زیر مثالی را می بینید برای تنظیم رمزگزاری **UTF-8**

```

<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="author" content="Mahnaz Mohtashim" />
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>

```

برای ارائه‌ی کاراکترهای چینی سنتی به یک صفحه‌ی استاتیک، صفحه‌ی وب باید دارای یک برچسب **<meta>** باشد تا رمزگذاری **Big5** را تنظیم کند.

```

<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="author" content="Mahnaz Mohtashim" />
<meta http-equiv="Content-Type" content="text/html; charset=Big5" />
</head>

```

```
<body>
  <p>Hello HTML5!</p>
</body>
</html>
```

آموزش استفاده از comment در html

کامنت قطعه ای از کد می باشد که توسط مرورگرها نادیده گرفته می شود. افزودن کامنت به کد HTML خود، تمرين خوبی است، به ویژه در داکیومنت های پیچیده برای نشان دادن بخش هایی از یک داکیومنت و یادداشت هایی دیگر به هر کس که کد را نگاه می کند. کامنت ها به شما و دیگران کمک می کنند تا کد خود را متوجه شوید و قابلیت خواندن آن را افزایش می دهد.

کامنت های HTML بین برچسب های `<!-- ... -->` واقع می شوند. بتایراین هر محتوای که بین این برچسب ها واقع شود، مثل کامنت با آن رفتار خواهد شد و توسط مرورگرها به طور کامل نادیده گرفته خواهد شد.



مثال

```
<!DOCTYPE html>
<html>
<head>
  <!-- Document Header Starts -->
  <title>This is document title</title>
</head><!-- Document Header Ends -->
<body>
  <p>Document content goes here.....</p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را بدون نمایش محتوای ارائه شده به عنوان بخشی از کامنت‌ها، به دنبال دارد.

Document content goes here.....

کامنت های معتبردر مقابل کامنت های نامعتبر

کامنت‌ها تودرتو نمی شوند، یعنی اینکه یک کامنت نمی تواند در داخل کامنت دیگری قرار بگیرد. خط تیره‌ی (dash) `--` ممکن نیست در داخل یک کامنت ظاهر شود، به جز در مواردی که بخشی از برچسب closing باشد. شما باید مطمئن باشید که هیچ فضایی در ابتدای رشته‌ی کامنت وجود ندارد.

مثال

در اینجا کامنت ارائه شده یک کامنت معتبر می باشد و توسط مرورگر پاک می شود.

```
<!DOCTYPE html>
<html>
<head>
  <title>Valid Comment Example</title>
</head>
<body>
  <!-- This is valid comment -->
  <p>Document content goes here.....</p>
</body>
</html>
```

اما خط زیر یک کامنت معتبر نیست و توسط مرورگر نمایش داده خواهد شد. این به این خاطر است که فضای بین حاشیه ی چپ پرانتز و علامت تعجب وجود دارد.

```
<!DOCTYPE html>
<html>
<head>
  <title>Invalid Comment Example</title>
</head>
<body>
  <!-- This is not a valid comment -->
  <p>Document content goes here.....</p>
</body>
</html>
```

<!-- This is not a valid comment -->

Document content goes here.....

کامنت های چندخطی

تاکنون فقط کامنت های تک خطی را مشاهده کردیم، اما **HTML** کامنت های چندخطی را نیز پشتیبانی می کند.

شما می توانید کامنت های چند خطی داشته باشید با استفاده از برچسب آغازگر **<--!** و پایان دهنده **-->** که قبل از اولین خط و در پایان آخرین خط قرار دارد، همانطور که در مثال زیر می بینید.

مثال

```

<!DOCTYPE html>
<html>
<head>
    <title>Multiline Comments</title>
</head>
<body>
    <!--
        This is a multiline comment and it can
        span through as many as lines you like.
    -->
    <p>Document content goes here.....</p>
</body>
</html>

```

این مثال نتیجه‌ی زیر را ایجاد خواهد کرد

Document content goes here.....

کامنت‌های شرطی

کامنت‌های شرطی تنها در اینترنت اکسپلورر روی ویندوز کار می‌کنند، اما توسط مرورگرهای دیگر نادیده گرفته می‌شوند. این کامنت‌ها در اکسپلورر 5 به بالاتر پشتیبانی می‌شوند و می‌توانند از آنها برای دادن دستورات شرطی به ورژن‌های مختلف IE استفاده کنید.

مثال

```

<!DOCTYPE html>
<html>
<head>
    <title>Conditional Comments</title>
    <!--[if IE 6]>
        Special instructions for IE 6 here
    <![endif]-->
</head>
<body>
    <p>Document content goes here.....</p>
</body>
</html>

```

در جایی که نیاز خواهید داشت یک صفحه‌ی طراحی متفاوت بر اساس ورژن‌های مختلف اینترنت اکسپلورر به کار بگیرید، به راه حلی نیاز خواهید داشت. در این موقع چنین کامنت‌های شرطی مفید خواهند بود.

استفاده از برچسب کامنت

مروگرها کمی هستند که استفاده از برچسب `<comment>` را برای کامنت بخشی از کد `HTML` استفاده می‌کنند.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Using Comment Tag</title>
</head>
<body>
  <p>This is <comment>not</comment> Internet Explorer.</p>
</body>
</html>
```

اگر از `IE` استفاده می‌کنید، نتیجه‌ی زیر حاصل خواهد شد

This is Internet Explorer.

اما اگر در حال استفاده از `IE` نمی‌باشد، نتیجه‌ی زیر حاصل می‌شود.

This is Internet Explorer.

کد اسکریپت کامنت

گرچه شما جاوا اسکریپت را با `HTML` یاد خواهید گرفت، اما در یک آموزش مجزا. در اینجا باید دقت کنید که اگر در حال استفاده از `html` در کد `vb script` یا `java script` خود هستید، بنابراین توصیه می‌شود که آن کد اسکریپت را در داخل کامنت‌های مناسب `HTML` قرار دهید طوریکه مروگرها قدمی بتوانند به درستی کار کنند.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Commenting Script Code</title>
  <script>
<!-- document.write("Hello World!")</pre>
```

```
//-->
</script>
</head>
<body>
<p>Hello , World!</p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را در پی خواهد داشت

Hello World!

Hello , World!

کامنت صفحات طراحی

گرچه شما **HTML** را در یک آموزش مجزا با صفحات طراحی فرا می‌گیرید، اما در اینجا به باید دقیق داشته باشید که اگر از **CSS** در کد **HTML** خود استفاده می‌کنید، بتایراین توصیه می‌شود که کد صفحه‌ی طراحی را داخل کامنت‌های مناسب **HTML** قرار دهید، طوریکه مرورگرهای قدیمی بتوانند کار کنند.

مثال

```
<!DOCTYPE html>
<html>
<head>
<title>Commenting Style Sheets</title>
<style>
<!--
.example {
    border :1px solid #4a7d49;
}
// -->
</style>
</head>
<body>
<div class="example">Hello , World!</div>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید می‌کند

Hello , World!

وارد کردن تصاویر در صفحات **HTML**

در این آموزش چگونگی استفاده از تصاویر در صفحات **HTML** را فرا می گیریم. زیبا سازی تصاویر و همچنین ترسیم بسیاری از مفاهیم پیچیده به یک روش ساده روی صفحه‌ی وب شما، بسیار مهم می باشد. این آموزش مراحل ساده‌ی استفاده از تصاویر در صفحات وب را به شما آموزش خواهد داد.

وارد کردن تصویر

شما می توانید با استفاده از برچسب **** هر تصویری را وارد صفحه‌ی وب خود کنید. در زیر ترکیب ساده‌ی استفاده از این برچسب را می بینید

```

```

برچسب **** یک برچسب خالی می باشد، یعنی تنها می تواند دارای لیستی از ویژگی‌ها باشد و دارای برچسب **closing** نمی باشد.

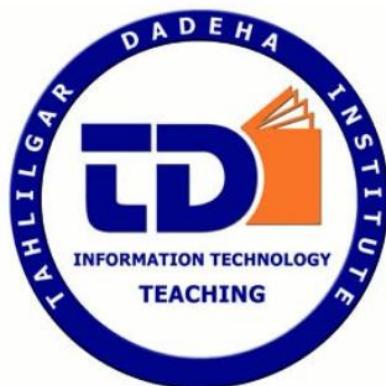
مثال برای امتحان کردن مثال، زیر اجازه بدھیم فایل **html** خود یعنی **test.htm** و فایل تصویر خود یعنی **test.png** را در یک مسیر قرار دهیم.

```
<!DOCTYPE html>
<html>
<head>
  <title>Using Image in Webpage</title>
</head>
<body>
  <p>Simple Image Insert</p>
  
</body>
</html>
```

این مثال نتیجه‌ی زیر را به دنبال خواهد داشت

Simple Image Insert

Simple Image Insert



DADEHA

INSTITUTE

شما می توانید از فایل تصویر **JPEG**, **GIF** یا **PNG** متناسب با راحتی خود استفاده کنید، اما مطمئن شوید که در ویژگی **src** نام فایل تصویر را به درستی وارد کردید. نام تصویر همیشه یک مورد هوشمند می باشد. ویژگی **alt** یک ویژگی می باشد که اگر تصویر نمایش داده نشود، یک متن جایگزین را برای آن مشخص می کند.

تنظیم موقعیت تصویر

معمولا ما تمام تصاویر خود را در یک مسیر مجزا قرار می دهیم. بنابراین اجازه بدھید فایل **test.htm** مربوط به **test.png** را در مسیر اصلی نگه داشته و یک مسیر فرعی **images** داخل مسیر اصلی، جایی که تصویر **HTML** را نگهداری می کنیم، ایجاد کنیم.

با فرض اینکه موقعیت تصویر ما "**image/test.png**" می باشد، مثال زیر را امتحان کنید

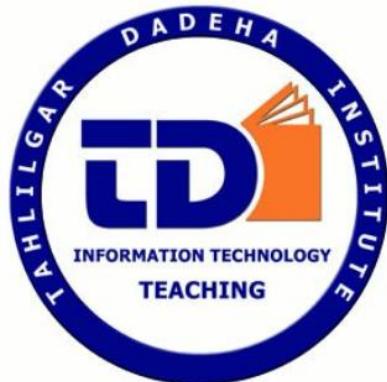
مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Using Image in Webpage</title>
</head>
<body>
  <p>Simple Image Insert</p>
  
</body>
</html>
```

این مثال نتیجه‌ی زیر را به دنبال دارد

Simple Image Insert

Simple Image Insert



DADEHA

INSTITUTE

تنظیم طول و عرض تصویر

می توانید طول و عرض تصویر را براساس نیاز خود و با استفاده از ویژگی های `height` و `width` تنظیم کنید.
شما می توانید طول و عرض تصویر را به پیکسل یا براساس درصد اندازه ی واقعی آن تنظیم کنید.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Set Image Width and Height</title>
</head>
<body>
  <p>Setting image width and height</p>
  
</body>
</html>
```

این مثال نتیجه ی زیر را در برخواهد داشت

Setting image width and height

Setting image width and height



تنظیم حاشیه‌ی تصویر

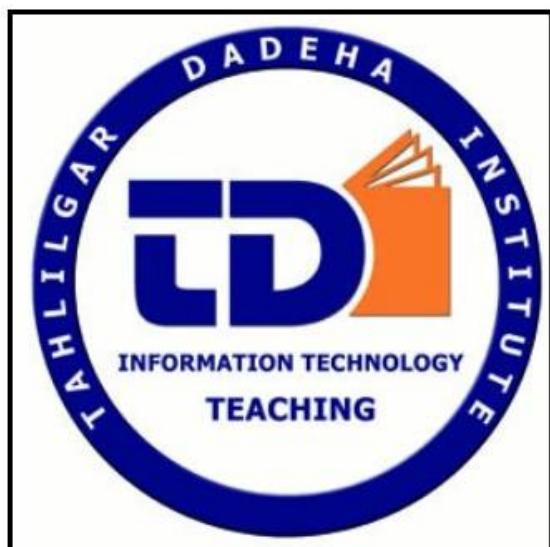
به طور پیش فرض تصویر حاشیه‌ای در اطراف خود خواهد داشت، شما می‌توانید ضخامت این حاشیه را با استفاده از ویژگی **border** و به واحد پیکسل تنظیم کنید. ضخامت ۰ یعنی هیچ حاشیه‌ای در اطراف تصویر وجود ندارد.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Set Image Border</title>
</head>
<body>
  <p>Setting image Border</p>
  
</body>
</html>
```

Setting image Border

این مثال نتیجه‌ی زیر را تولید خواهد کرد



تنظیم همترازی تصویر

به طور پیش فرض تصویر در سمت چپ صفحه تنظیم می شود، اما می توانید از ویژگی **align** برای تنظیم تصویر در سمت راست یا مرکز صفحه استفاده کنید.

مثال

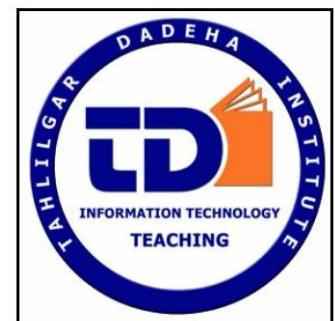
```
<!DOCTYPE html>
<html>
<head>
  <title>Set Image Alignment</title>
</head>
<body>
  <p>Setting image Alignment</p>
  
</body>
</html>
```

Setting image Alignment

این مثال نتیجه‌ی زیر را به دنبال دارد



Setting image Alignment



آموزش Table در HTML

جدول های **HTML** به نویسنده‌گان وب اجازه می دهند تا داده‌های مانند متن، تصاویر، لینک‌ها، جداول های دیگر و غیره در ردیف‌ها و ستون‌ها تنظیم کنند. جدول های **HTML** با استفاده از برچسب **<table>** ایجاد می شوند که در آن برچسب **<tr>** برای ایجاد ردیف‌ها و برچسب **<td>** برای ایجاد داده‌های سلول‌ها استفاده می شوند.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Tables</title>
</head>
<body>
  <table border="1">
    <tr>
      <td>Row 1, Column 1</td>
      <td>Row 1, Column 2</td>
    </tr>
    <tr>
      <td>Row 2, Column 1</td>
      <td>Row 2, Column 2</td>
    </tr>
  </table>
</body>
</html>
```

در اینجا **border** یک ویژگی از برچسب **<table>** می باشد و برای قرار دادن حاشیه در پهنهای همه سلول ها استفاده می شود. اگر به حاشیه احتیاج نداشته باشید می توانید از **border="0"** استفاده کنید.

تیتر جدول

تیتر جدول می تواند با استفاده از برچسب **<th>** تعریف شود. این برچسب برای جایگزینی با برچسب **<td>** استفاده می شود که برچسب **<td>** برای نمایش داده حقیقی سلول استفاده می شود. طبیعتاً شما ردیف های بالای جدول را به عنوان تیتر قرار می دهید.

ویژگی های **cellspacing** و **cellpadding**

دو ویژگی به نام های **Cellspacing** و **Cellpadding** وجود دارند که برای تنظیم محیط سفید در سلول های جدول استفاده می شوند. ویژگی **cellspacing** عرض حاشیه را تعریف می کند، در حالیکه **cellpadding** فاصله بین حاشیه های سلول و محتوای آن را تعریف می کند.

مثال تیتر جدول و **cellspacing** و **cellpadding**

```
<!DOCTYPE html>
<html>
<head>
```

```

<title>HTML Table Cellpadding</title>
</head>
<body>
<table border="1" cellpadding="5" cellspacing="5">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
<td>Ramesh Raman</td>
<td>5000</td>
</tr>
<tr>
<td>Shabbir Hussein</td>
<td>7000</td>
</tr>
</table>
</body>
</html>

```

ویژگی های rowspan و colspan

اگر بخواهید دو یا بیشتر از دو ستون را با هم تلفیق کنید از ویژگی **colspan** استفاده می کنید. به طور مشابه اگر بخواهید دو یا بیشتر از دو ردیف را در یک ردیف تلفیق کنید از **rowspan** استفاده کنید.

مثال

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table Colspan/Rowspan</title>
</head>
<body>
<table border="1">
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
<tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row 1 Cell 3</td></tr>
<tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr>
<tr><td colspan="3">Row 3 Cell 1</td></tr>
</table>
</body>
</html>

```

پس زمینه‌ی جدول

می‌توانید به دو روش زیر برای جدول خود زمینه‌ای تنظیم کنید.

ویژگی **Bg color** می‌توانید رنگ زمینه را برای همه‌ی جدول و یا تنها برای یک سلول تنظیم کنید.

ویژگی **background** می‌توانید یک تصویر را برای کل جدول یا تنها یک سلول تنظیم کنید.

همچنین می‌توانید با استفاده از ویژگی **bordercolor** رنگ حاشیه را نیز تنظیم کنید.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Table Background</title>
</head>
<body>
  <table border="1" bordercolor="green" bgcolor="yellow">
    <tr>
      <th>Column 1</th>
      <th>Column 2</th>
      <th>Column 3</th>
    </tr>
    <tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row 1 Cell 3</td></tr>
    <tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr>
    <tr><td colspan="3">Row 3 Cell 1</td></tr>
  </table>
</body>
</html>
```

در اینجا مثالی در رابطه با استفاده از ویژگی **background** می‌بینید. در این مثال ما از تصویری موجود در **directory image** استفاده کرده‌ایم.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Table Background</title>
</head>
<body>
  <table border="1" bordercolor="green" background="/images/test.png">
    <tr>
      <th>Column 1</th>
      <th>Column 2</th>
      <th>Column 3</th>
    </tr>
    <tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row 1 Cell 3</td></tr>
    <tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr>
  </table>
</body>
</html>
```

```

<tr><td colspan="3">Row 3 Cell 1</td></tr>
</table>
</body>
</html>

```

طول و عرض جدول

شما می توانید طول و عرض جدولی را با استفاده از ویژگی های **height** و **width** تنظیم کنید. شما می توانید طول و عرض جدول را به پیکسل و یا متناسب با درصد صفحه ی جاری تنظیم کنید.

مثال

```

<!DOCTYPE html>
<html>
<head>
  <title>HTML Table Width/Height</title>
</head>
<body>
  <table border="1" width="400" height="150">
    <tr>
      <td>Row 1, Column 1</td>
      <td>Row 1, Column 2</td>
    </tr>
    <tr>
      <td>Row 2, Column 1</td>
      <td>Row 2, Column 2</td>
    </tr>
  </table>
</body>
</html>

```

آموزشگاه تحلیلکرداری

شرح جدول

برچسب **caption** یک توضیح یا یک تیتر برای جدول می باشد که در بالای جدول نمایش داده می شود. این برچسب در ورژن های جدید **HTML/XHTML** توصیه می شود.

مثال

```

<!DOCTYPE html>
<html>
<head>
  <title>HTML Table Caption</title>
</head>
<body>
  <table border="1" width="100%">
    <caption>This is the caption</caption>
    <tr>
      <td>row 1, column 1</td>

```

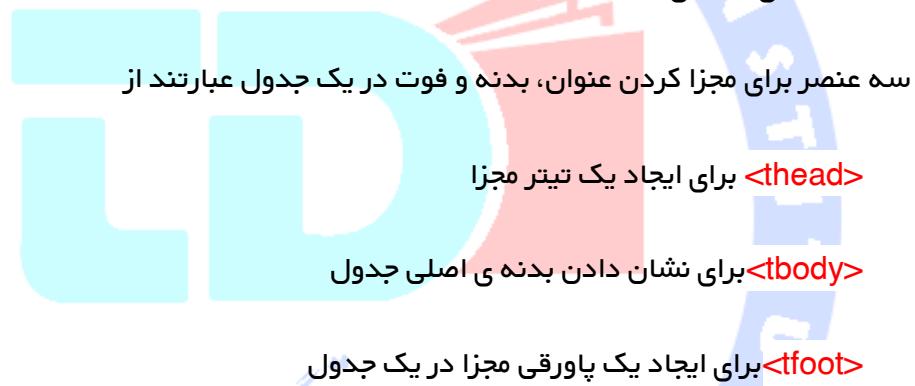
```

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

```

تیتر، بدن و پاورقی جدول

جدول ها می توانند به سه بخش تقسیم شوند تیتر، بدن و پاورقی جدول. عنوان و فوت (***tfoot***) در واقع شبیه به تیتر و پاورقی در یک فایل پردازش **word** می باشند که برای هر صفحه یکی می باشد، در حالیکه بدن همان نگه دارنده ای محتوای اصلی جدول می باشد.



یک جدول برای نشان دادن صفات و گروه های مختلف یک داده، ممکن است دارای عناصر **<tbody>** مختلفی باشد، اما ظاهر شدن برچسب های **<thead>** و **<tbody>** قبل از **<tfoot>** زیاد مهم نیست.

مثال

```

<!DOCTYPE html>
<html>
<head>
  <title>HTML Table</title>
</head>
<body>
  <table border="1" width="100%">
    <thead>
      <tr>
        <td colspan="4">This is the head of the table</td>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td colspan="4">This is the foot of the table</td>
      </tr>
    </tbody>
  </table>
</body>
</html>

```

```

</tr>
</tfoot>
<tbody>
<tr>
<td>Cell 1</td>
<td>Cell 2</td>
<td>Cell 3</td>
<td>Cell 4</td>
</tr>
</tbody>
</table>
</body>
</html>

```

جدول های تودرتو

شما می توانید از یک جدول در داخل جدولی دیگر استفاده کنید. نه تنها جدول ها، بلکه می توانید تمام برچسب

ها را در داخل برچسب داده **<td>** استفاده کنید.

مثال

در زیر **مثال** را از استفاده ی یک جدول و برچسب های دیگر در داخل یک سلول مشاهده می کنید.

```

<!DOCTYPE html>
<html>
<head>
<title>HTML Table</title>
</head>
<body>
<table border="1" width="100%">
<tr>
<td>
<table border="1" width="100%">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
<td>Ramesh Raman</td>
<td>5000</td>
</tr>
<tr>
<td>Shabbir Hussein</td>
<td>7000</td>
</tr>
</table>
</td>
</tr>
</table>
</body>
</html>

```

آموزش List

سه روش برای مشخص کردن لیست هایی از اطلاعات ارائه می دهد. تمام لیست ها باید شامل یک یا بیشتر از یک عنصر باشند. لیست ها ممکن است شامل موارد زیر شوند.

- لیست بدون ترتیب. این لیست آیتم ها را با استفاده از **bullet** های ساده لیست می کند.

- لیست منظم. این لیست از نمودارهای عددی مختلف برای لیست کردن آیتم های شما استفاده می کند.

- لیست تعریف. این لیست آیتم های شما را به همان روشی که در دیکشنری منظم شده اند، منظم می کند.

لیست های بدون ترتیب

لیست بدون ترتیب مجموعه ای از آیتم های مربوط به هم می باشد که هیچگونه نظم و ترتیب خاصی ندارند. این لیست با استفاده از برجسب **** در **HTML** ایجاد می شود. هر آیتم در لیست با یک **bullet** مشخص می شود.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Unordered List</title>
</head>
<body>
<ul>
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
</ul>
</body>
</html>
```

ویژگی type

می توانید از ویژگی **type** برای برجسب **** استفاده کنید تا نوع **bullet** خود را مشخص کنید، که به طور پیش فرض یک دیسک می باشد. در زیر گزینه های ممکن را مشاهده می کنید.

```
<ul type="square">  
<ul type="disc">  
<ul type="circle">
```

در زیر مثالی را میبینید که در آن از **<ul type="square">** استفاده می کنیم.

```
<!DOCTYPE html>  
<html>  
<head>  
<title>HTML Unordered List</title>  
</head>  
<body>  
<ul type="square">  
<li>Beetroot</li>  
<li>Ginger</li>  
<li>Potato</li>  
<li>Radish</li>  
</ul>  
</body>  
</html>
```



در زیر مثالی را می بینید که در آن از **<ul type="disc">** استفاده کرده ایم.

```
<!DOCTYPE html>  
<html>  
<head>  
<title>HTML Unordered List</title>  
</head>  
<body>  
<ul type="disc">  
<li>Beetroot</li>  
<li>Ginger</li>  
<li>Potato</li>  
<li>Radish</li>  
</ul>  
</body>  
</html>
```

در زیر مثالی را می بینید که در آن از **<ul type="circle">** استفاده کرده ایم.

```
<!DOCTYPE html>  
<html>  
<head>  
<title>HTML Unordered List</title>  
</head>  
<body>
```

```
<ul type="circle">
<li>Beetroot</li>
<li>Ginger</li>
<li>Potato</li>
<li>Radish</li>
</ul>
</body>
</html>
```

لیست های منظم HTML

اگر تمایل دارید آیتم های خود را به جای قرار دادن در یک لیست دارای bullet، در یک لیست عددگذاری شده قرار دهید، می توانید از لیست منظم HTML استفاده کنید. این لیست با استفاده از برچسب ایجاد می شود. شماره گذاری از یک شروع شده و برای هر لیست منظم عنصر بعدی با اضافه شدن یک عدد و به همراه برچسب اضافه می شود.

می توانید از ویژگی type برای برچسب استفاده کنیم تا نوع شماره گذاریمورد نظر خود را مشخص کنید. به طور پیش فرض شماره گذاری به وسیله‌ی عدد انجام می شود. در زیر گزینه‌های ممکن را مشاهده می کنید.

```
<ol type="1">
- Default-Case Numerals.
<ol type="I">
- Upper-Case Numerals.
<ol type="i">
- Lower-Case Numerals.
<ol type="a">
- Lower-Case Letters.
<ol type="A"> - Upper-Case Letters.
```

در زیر مثالی را می بینید که در آن از <ol type="1"> استفاده کرده ایم.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
<ol type="1">
<li>Beetroot</li>
<li>Ginger</li>
<li>Potato</li>
<li>Radish</li>
</ol>
</body>
</html>
```

در زیر مثالی را می بینید که در آن از <ol type="I"> استفاده می کنیم.

```
<!DOCTYPE html>
<html>
<head>
    <title>HTML Ordered List</title>
</head>
<body>
    <ol type="I">
        <li>Beetroot</li>
        <li>Ginger</li>
        <li>Potato</li>
        <li>Radish</li>
    </ol>
</body>
</html>
```

در زیر مثالی را میبینید که در آن از **<ol type="i">** استفاده کرده ایم.

```
<!DOCTYPE html>
<html>
<head>
    <title>HTML Ordered List</title>
</head>
<body>
    <ol type="i">
        <li>Beetroot</li>
        <li>Ginger</li>
        <li>Potato</li>
        <li>Radish</li>
    </ol>
</body>
</html>
```

در زیر مثالی را می بینید که در آن از **<ol type="A">** استفاده کرده ایم.

```
<!DOCTYPE html>
<html>
<head>
    <title>HTML Ordered List</title>
</head>
<body>
    <ol type="A">
        <li>Beetroot</li>
        <li>Ginger</li>
        <li>Potato</li>
        <li>Radish</li>
    </ol>
</body>
</html>
```

در زیر مثالی را می بینید که در آن از **<ol type="a">** استفاده کرده ایم.

```
<!DOCTYPE html>
```

```

<html>
<head>
  <title>HTML Ordered List</title>
</head>
<body>
<ol type="a">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
</ol>
</body>
</html>

```



شما می توانید از ویژگی **start** برای برچسب **** استفاده کنید تا نقطه‌ی شروع شماره گذاری خود را مشخص کنید. در زیر گزینه‌های ممکن را مشاهده می کنید.

```

<ol type="1" start="4">
  - Numerals starts with 4.
<ol type="I" start="4">
  - Numerals starts with IV.
<ol type="i" start="4">
  - Numerals starts with iv.
<ol type="a" start="4">
  - Letters starts with d.
<ol type="A" start="4">
  - Letters starts with D.

```

در زیر مثالی را می بینید که در آن از **<ol type="i" start="4">** استفاده می کنیم.

```

<!DOCTYPE html>
<html>
<head>
  <title>HTML Ordered List</title>
</head>
<body>
<ol type="i" start="4">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
</ol>
</body>
</html>

```

لیست های تعریف HTML

لیست هایی به نام لیست های تعریف را پشتیبانی می کنند که در این لیست ها ورودی ها مانند ترتیب لغات در دیکشنری قرار می گیرند. این لیست یک روش ایده آل برای ارائه یک فهرست از معانی یا لیستی از اصطلاحات یا لیستی از نام ها و مقادیر می باشد.

لیست تعریف از سه برجسب زیر استفاده می کند.

- شروع لیست را تعریف می کند.

- یک عبارت

- تعریف عبارت

- پایان لیست را تعریف می کند.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Definition List</title>
</head>
<body>
  <dl>
    <dt><b>HTML</b></dt>
    <dd>This stands for Hyper Text Markup Language</dd>
    <dt><b>HTTP</b></dt>
    <dd>This stands for Hyper Text Transfer Protocol</dd>
  </dl>
</body>
</html>
```

لينک های متن HTML

یک صفحه‌ی وب می‌تواند لینک‌های متنوعی داشته باشد که شما را مستقیماً به صفحات دیگر یا حتی بخش‌هایی خاص از یک صفحه‌ی ارائه شده می‌برد. این لینک‌ها هایپرلینک نامیده می‌شوند.

هایپرلینک ها به بازدیدکنندگان اجازه می دهند تا با کلیک کردن روی لغات، اصطلاحات و تصاویر بین صفحات وب مسیریابی کنند. شما می توانید هایپرلینک ها را روی صفحه ی وب با استفاده از متن یا تصاویر موجود ایجاد کنید.

لینک کردن داکیومت ها

یک لینک با استفاده از برچسب `<a>` در HTML تعیین می شود. این برچسب anchor tag نامیده می شود و هر چیزی بین برچسب آغازین `<a>` و پایانی `` بخشی از لینک می شود و یک یوزر می تواند آن را کلیک کرده تا به داکیومنت لینک شده برسد. در زیر ترکیب ساده ای استفاده از برچسب `<a>` را می بینید.

```
<a href="Document URL" ... attributes-list>Link Text</a>
```

مثال

اجازه دهید مثال زیر را امتحان کنیم که <http://www.tahlildadeh.com> را در صفحه ی شما لینک می کند.

```
<!DOCTYPE html>
<html>
<head>
  <title>Hyperlink Example</title>
</head>
<body>
  <p>Click following link</p>
  <a href="http://www.tahlildadeh.com" target="_self">tahlildadeh</a>
</body>
</html>
```

این مثال نتیجه ی زیر را تولید خواهد کرد که شما می توانید روی لینک تولید شده ی [tahlildadeh](http://www.tahlildadeh.com) کلیک کنید تا به صفحه ی اصلی [tahlildadeh](http://www.tahlildadeh.com) برسید.

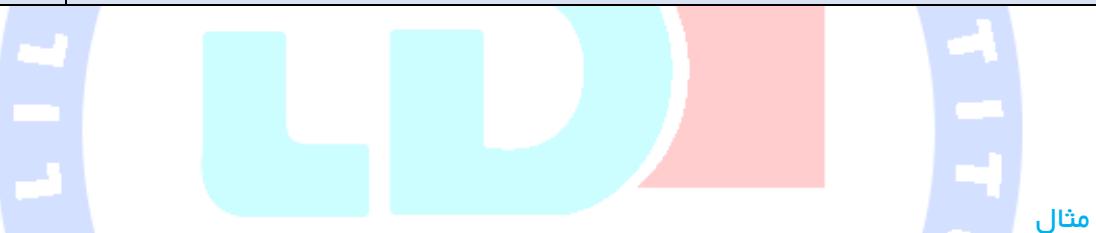
Click following link

[Tahlildadeh](http://www.tahlildadeh.com)

ویژگی target

ما از ویژگی target در مثال قبلی خود استفاده کردیم . این ویژگی برای مشخص کردن موقعیتی که داکیومنت لینک شده باز می شود، مورد استفاده قرار می گیرد. در زیر گزینه های ممکن را مشاهده می کنید.

Option	Description
_blank	دایرکت لینک شده را در یک پنجره یا تب جدید باز می کند.
_self	دایرکت لینک شده را در همان چارچوب باز می کند.
_parent	دایرکت لینک شده را در چارچوب اصلی باز می کند.
_top	دایرکت لینک شده در کل بدنه ی پنجره باز می شود.
targetframe	دایرکت لینک شده را در یک targetframe نام گذاری شده باز می کند.



مثال

برای درک تفاوت اصلی در برخی گزینه های ارائه شده در ویژگی **target** زیر را امتحان کنید.

```
<!DOCTYPE html>
<html>
<head>
  <title>Hyperlink Example</title>
  <base href="http://www.tahlildadeh.com/">
</head>
<body>
  <p>Click any of the following links</p>
  <a href="http://www.tahlildadeh.com" target="_blank">Opens in New</a> |
  <a href="http://www.tahlildadeh.com" target="_self">Opens in Self</a> |
  <a href="http://www.tahlildadeh.com" target="_parent">Opens in Parent</a> |
  <a href="http://www.tahlildadeh.com" target="_top">Opens in Body</a>
</body>
```

این مثال نتیجه‌ی زیر را تولید خواهد کرد که می توانید روی لینک‌های مختلف کلیک کنید تا تفاوت بین گزینه های مختلف ارائه شده برای ویژگی **target** را درک کنید.

Click any of the following links

[Opens in New](http://www.tahlildadeh.com) | [Opens in Self](#) | [Opens in Parent](#) | [Opens in Body](#)

استفاده از مسیر پایه

وقتی که شما داکیومنت های **HTML** را مناسب با همان وب سایت لینک می کنید، ارائه ی یک **URL** کامل برای هر لینک ضروری نیست. اگر از برچسب **<base>** در تیتر داکیومنت **HTML** خود استفاده می کنید، می توانید از دست آن خلاص شوید. این برچسب برای ارائه ی یک مسیر پایه برای همه ی لینک ها استفاده می شود. بنابراین مرورگر شما مسیر ارائه شده ی مرتبط را به مسیر پایه پیوند خواهد داد و یک **URL** کامل ایجاد خواهد کرد.

مثال

مثال زیر از برچسب **<base>** برای مشخص کردن **URL** پایه استفاده می کند و پس از آن ما می توانیم به جای ارائه ی **URL** کامل برای هر لینک از مسیرهای مرتبط ایتفاچ کنیم.

```
<!DOCTYPE html>
<html>
<head>
  <title>Hyperlink Example</title>
  <base href="http://www.tahlildadeh.com/">
</head>
<body>
  <p>Click following link</p>
  <a href="/html/index.htm" target="_blank">HTML tahlildadeh</a>
</body>
</html>
```

این **مثال** نتیجه ی زیر را تولید خواهد کرد که می توانید روی لینک تولید شده ی **HTML tahlildadeh** کلیک کنید تا به آموزش **HTML** برسید.

اکنون **URL** ارائه شده ی **a href="/html/index.htm"** با عنوان

در نظر گرفته می شود.

Click following link

HTML tahlildadeh

لینک شدن به بخشی از صفحه

می توانید برای بخش خاصی از صفحه‌ی وب ارائه شده، با استفاده از ویژگی **name** یک لینک ایجاد کنید. این امر یک فرایند دو مرحله‌ای می‌باشد.

نخست اینکه در مکانی که می‌خواهید به داخل صفحه‌ی وب برسید یک لینک ایجاد کنید و آن را با استفاده از **برچسب <a>...** نام‌گذاری کنید.

```
<h1>HTML Text Links <a name="top"></a></h1>
```

مرحله‌ی دوم ایجاد یک هایپر لینک می‌باشد برای لینک کردن داکیومنت و قرار دادن در مکانی که می‌خواهید برسید.

```
<a href="/html/html_text_links.htm#top">Go to the Top</a>
```

و این مثال لینک زیر را تولید خواهد کرد که می‌توانید در آن روی لینک تولید شده‌ی **Go to the Top** کلیک کنید تا به نقطه‌ی بالای آموزش **HTML Text Link** برسید.

Go to the Top

تنظیم رنگ‌های لینک

شما می‌توانید رنگ لینک‌های خود، لینک‌های فعال و لینک‌های مشاهده شده را با استفاده از ویژگی‌های **link** و **vlink** و **alink** از برچسب **<body>** تنظیم کنید.

مثال

مثال زیر را در **test.htm** ذخیره کنید و سپس می‌توانید آن را در هر مرورگری باز کنید تا ببینید که چگونه ویژگی‌های **vlink** و **alink** کار می‌کنند.

```
<!DOCTYPE html>
<html>
<head>
  <title>Hyperlink Example</title>
  <base href="http://www.tahlidadeh.com/">
</head>
<body alink="#54A250" link="#040404" vlink="#F40633">
  <p>Click following link</p>
```

```
<a href="/html/index.htm" target="_blank">HTML tahlildadeh</a>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید می‌کند. فقط رنگ لینک را قبل از کلیک کردن روی آن چک کنید، سپس رنگ ان را در هنگام فعال کردن و بازدید آن چک کنید.

Click following link

HTML tahlildadeh

دانلود کردن لینک‌ها

شما می‌توانید لینک متنی ایجاد کنید تا فایل‌های قابل دانلود **DOC**, **ZIP** و **PDF** خود را بسازید. این کار بسیار ساده می‌باشد، فقط کافیست یک **URL** کامل از فایل قابل دانلود ارائه بدهید.

```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
</head>
<a href="http://tahlildadeh.com/Files/Articles/04GL07.pdf">Download PDF File</a>
</body>
</html>
```

این مثال لینک زیر را تولید خواهد کرد و برای دانلود یک فایل استفاده می‌شود.

Download PDF File

آموزش Link

تاکنون مشاهده کردیم که چگونه با استفاده از متن، یک لینک هایپرلینک است ایجاد کنیم و همچنین فرا گرفته ایم که چگونه از تصاویر در صفحات وب خود استفاده کنیم. اکنون فرا خواهیم گرفت که چگونه با استفاده از تصاویر هایپرلینک ایجاد کنیم.

مثال

استفاده از تصویر به عنوان هایپرلینک بسیار ساده می‌باشد. لازم است که یک تصویر را در داخل هایپرلینک در محل تصویر قرار دهیم، همانطور که در زیر نشان داده شده است.

```
<!DOCTYPE html>
```

66

```
<html>
<head>
  <title>Image Hyperlink Example</title>
</head>
<body>
  <p>Click following link</p>
  <a href="http://www.tahlildadeh.com" target="_self">
    
  </a>
</body>
</html>
```

این ساده ترین راه ایجاد هایپرلینک با استفاده از تصاویر می باشد.

تصاویر حساس به ماوس

استانداردهای **XHTML** و **HTML** یک ویژگی ارائه می دهند که به شما اجازه می دهد لینک های مختلفی را در داخل یک تصویر اجرا کنید. شما می توانید براساس مختصات مختلف موجود روی تصویر، لینک های مختلفی را روی یک تصویر مجزا ایجاد کنید. زمانی که لینک های متفاوت به مختصات متفاوت ضمیمه می شود، می توانیم برای باز کردن داکیومنت های تارگت روی بخش های مختلف تصویر کلیک کنیم. چنین تصاویری که به ماوس حساس می باشند، تصاویر نقشه نامیده می شوند. دو روش برای ایجاد چنین تصاویری وجود دارد.

– توسط ویژگی **ismap** از برچسب **** فعال می شود و دستیابی به یک سرور و پردازش برنامه های مربوط به تصویر نقشه لازم می باشد.

– با استفاده از ویژگی **usemap** از برچسب **** همراه با برچسب های **<area>** و **<map>** ایجاد می شود.

Server side image maps

در اینجا به سادگی تصویر خود را داخل یک هایپرلینک قرار داده و از ویژگی **ismap** استفاده کنید که آن را یک تصویر خاص می سازد و وقتی یوزر در بخشی از تصویر کلیک می کند، مرورگر مختصات اشاره گر ماوس را همراه با **URL** مشخص شده در برچسب **<a>** به سرور وب منتقل می کند. سرور از مختصات اشاره گر ماوس استفاده می کند تا تعیین کند کدام داکیومنت باید به مرورگر بازگردانده شود.

زمانی که **ismap** استفاده می شود، ویژگی **href** از برجسب **URL** یک برنامه‌ی سرور را مانند یک **cgi** یا اسکریپت **PHP** و غیره در برداشته باشد، تا درخواست ورودی را براساس مختصات انتقال داده شده پردازش کند. مختصات موقعیت ماوس پیکسل های صفحه میباشند که از گوشه‌ی بالای سمت چپ تصویر شمرده می شوند و با **(0,0)** شروع می شوند. مختصات دنبال شده با یک علامت سوال، به انتها **URL** اضافه می شوند.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>ISMAP Hyperlink Example</title>
</head>
<body>
  <p>Click following link</p>
  <p>Click following link</p>
  <a href="Exp1.html" target="_self">
    
  </a>
</body>
</html>
```

سپس مرورگر پارامترهای جستجوی زیر را به سرور می فرستد که می تواند توسط اسکریپت **ismap.cgi** یا فایل **map** پردازش شود و شما می توانید هر داکیومنتی را که دوست داشته باشید به این مختصات لینک کنید.

/cgi-bin/ismap.cgi?20,30

از این طریق شما می توانید لینک های مختلفی را به مختصات متفاوت یک تصویر اختصاص دهید، و وقتی این مختصات کلیک می شوند، می توانید داکیومنت های لینک شده‌ی منتظر را باز کنید.

توجه

شما برنامه نویسی **CGI** را زمانی فراخواهید گرفت که برنامه نویسی **perl** را مطالعه کنید. می توانید اسکریپت خود را بنویسید تا این مختصات انتقال داده شده را با استفاده از **PHP** یا هر اسکریپت دیگری پردازش کنید. فعلًا اجازه بدھید روی یادگیری **HTML** تمرکز کنیم، می توانید این فصل را بعدا بازبینی کنید.

Client side image maps

این تصاویر به وسیله‌ی ویژگی `usemap` از برچسب `` فعال می‌شوند و به وسیله‌ی برچسب‌های ضمیمه `<area>` و `<map>` تعریف می‌شوند.

نقشه‌ای که قرار است نقشه را طراحی کند، به وسیله‌ی `` به عنوان یک تصویر عادی وارد صفحه می‌شود، به جز اینکه این مورد ویژگی اضافه‌ای به نام `usemap` همرا خود دارد. مقدار ویژگی `usemap` مقداری می‌باشد که در یک برچسب `<map>` استفاده می‌شود تا برچسب‌های تصویر و نقشه را لینک کند. `<map>` همراه با برچسب‌های `<area>` همه‌ی مختصات تصویر و لینک‌های مربوطه را تعریف می‌کند.

برچسب `<area>` در داخل برچسب نقشه مختصات و شکل حاشیه‌های قابل کلیک در داخل تصویر را تعریف می‌کند. در اینجا **مثال** از تصویر نقشه می‌بینید.

```
<!DOCTYPE html>
<html>
<head>
  <title>USEMAP Hyperlink Example</title>
</head>
<body>
  
  <map name="planetmap">
    <area shape="rect" coords="0,0,40,126" href="02.jpg">
    <area shape="circle" coords="90,58,20" href="04.jpg">
    <area shape="circle" coords="124,78,20" href="05.jpg">
  </map>
</body>
</html>
```



سیستم مختصات

مقدار حقیقی مختصات کانلا به شکل درسوال وابسته است. در اینجا خلاصه‌ای می‌بینید که قرار است با **مثال** های مفصل دنبال شوند.

`rect = x1 , y1 , x2 , y2`

`x1` و `y1` مختصات گوشی بالای سمت چپ از مستطیل می‌باشد. `x2` و `y2` مختصات گوشی سمت راست پایین می‌باشند.

`circle = xc , yc , radius`

`x` و `y` مختصات مرکز دایره و `radius` شعاع دایره می باشد. دایره ای به مرکز 200,50 با شعاع 25 دارای ویژگی `coords="200,50,25"` خواهد بود.

`poly = x1 , y1 , x2 , y2 , x3 , y3 , ... xn , yn`

جفت های مختلف `x-y` رئوس چند ضلعی می باشند، با یک خط که از یک نقطه به نقطه‌ی دیگر کشیده شده. یک چندضلعی لوزی شکل با بالاترین راس آن در نقطه‌ی 20,20 و 40 پیکسل، در عریض ترین نقطه‌ی خود دارای ویژگی `coords="20,20,40,40,20,60,0,40"` می باشد.

تمام مختصات مربوط به بالاترین گوشه‌ی سمت چپ تصویر می باشند. هر شکل دارای یک `URL` مربوطه می باشد. می توانید از هر نرم افزار تصویری برای دانستن مختصات موقعیت‌های مختلف استفاده کنید.

لینک ایمیل HTML

قرار دادن یک لینک ایمیل `HTML` روی صفحه‌ی وب خود کار سختی نیست، اما این کار ممکن است باعث بروز مشکل اسپم‌های غیرضروری در اکانت ایمیل شما شود. افرادی هستند که می توانند برنامه‌هایی را برای جمع آوری چنین ایمیل‌هایی اجرا کنند و سپس آنها را با راه‌های مختلف برای اسپم کردن استفاده کنند.

می توانید از گزینه دیگری استفاده کنید تا افراد به راحتی بتوانند به شما ایمیل ارسال کنند. یک گزینه می تواند استفاده از فرم‌های `HTML` باشد برای جمع آوری داده‌های یوزر و سپس استفاده از اسکریپت `PHP` یا `CGI` برای ارسال ایمیل.

برای یک **مثال** ساده، فرم `Contact Us` را چک کنید. با استفاده از این فرم یک فیدبک از یوزر گرفته و سپس از یک برنامه‌ی `CGI` استفاده می کنیم که در حال جمع آوری این اطلاعات و ارسال ایمیل به یک `email ID` ارائه شده می باشد.

برچسب HTML Email

برچسب `<a>` در `HTML` برای مشخص کردن یک آدرس ایمیل و ارسال ایمیل، گزینه‌هایی را به شما ارائه می دهد. در حالیکه از برچسب `<a>` به عنوان یک `email tag` استفاده می کنید، از `mailto:email address` همراه با ویژگی `href` نیز استفاده خواهید کرد. در زیر ترکیب استفاده از `mailto` را به جای استفاده از `http` می بینید.

```
<a href="mailto:abc@example.com">Send Email</a>
```

این کد لینک زیر را تولید خواهد کرد که می توانید به آن ایمیل ارسال کنید

[Send Email](#)

اگر یوزر روی این لینک کلیک کند، آغاز به کار خواهد کرد (مانند **outlook** و **lotus notes** **email client** و **express** و غیره). که روی کامپیوتر شما نصب شده است. ریسک دیگری در استفاده از این گزینه برای ارسال ایمیل وجود دارد، زیرا اگر یوزر **email client** نصب شده روی کامپیوتر خود نداشته باشد، ارسال ایمیل ممکن خواهد بود.

تنظیمات پیش فرض

شما می توانید یک موضوع ایمیل و یک بدنه ی ایمیل پیش فرض همراه با آدرس ایمیل خود ایجاد کنید. در زیر **مثال** استفاده از بدنه و موضوع پیش فرض را مشاهده می کنید.

```
<a href="mailto:abc@example.com?subject=Feedback&body=Message">  
    Send Feedback  
</a>
```

این کد نیز لینک زیر را تولید خواهد کرد که می توانید ایمیل ارسال کنید.

[Send Feedback](#)

آموزش HTML Layout

برای یک صفحه ی وب بسیار مهم است، چرا که دید بهتری نسبت به وب سایت شما ارائه می دهد. طراحی **Layout** خوب با منظره و حس خوب برای یک وب سایت زمان فوق العاده زیادی می گیرد. این روزها همه ی وب سایت های مدرن از چارچوب هایی بر اساس جاوا اسکریپت و **CSS** استفاده میکنند تا به وب سایت های پاسخگو و دینامیک پیروز شوند، اما کار یک **Layout** در صفحه ی وب شما، صرفا از **HTML** و ویژگی های ان استفاده می کند.

استفاده از جدول‌ها – HTML Layout

ساده‌ترین و محبوب‌ترین راه برای ایجاد **Layout** در **HTML** می‌باشد. این جدول‌ها در ردیف‌ها و ستون‌ها منظم‌می‌شوند که شما می‌توانید از این ردیف‌ها و ستون‌ها به هر طریقی که می‌خواهید استفاده کنید.

برای **مثال**، نمونه **HTML Layout** زیر از طریق استفاده‌ی یک جدول با سه ردیف و دو ستون به دست می‌آید، اما عنوان و پاورقی ستون هر دو ستون را با استفاده‌ی **colspan** احاطه می‌کند.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Layout using Tables</title>
</head>
<body>
  <table width="100%" border="0">
    <tr>
      <td colspan="2" bgcolor="#b5dcb3">
        <h1>This is Web Page Main title</h1>
      </td>
    </tr>
    <tr valign="top">
      <td bgcolor="#aaaa" width="50">
        <b>Main Menu</b><br />
        HTML<br />
        PHP<br />
        PERL...
      </td>
      <td bgcolor="#eeee" width="100" height="200">
        Technical and Managerial Tutorials
      </td>
    </tr>
    <tr>
      <td colspan="2" bgcolor="#b5dcb3">
        <center>
          Copyright © 2007 Tahlidadeh.com
        </center>
      </td>
    </tr>
  </table>
</body>
</html>
```



این **مثال** نتیجه‌ی زیر را تولید خواهد کرد.

This is Web Page Main title		
Main	Menu	Technical and Managerial Tutorials
HTML		
PHP		
PERL...		
Copyright © 2007 TahlilDadeh.com		

چند سه‌تونی – استفاده از جدول‌ها Layout

شما می‌توانید صفحه‌ی وب خود را طوری طراحی کنید تا محتوای وب خود را در چند صفحه قرار دهید. می‌توانید محتوای خود را در سه‌تون وسط قرار دهید و از سه‌تون سمت چپ برای قرار دادن منو و از سه‌تون سمت راست برای تبلیغ یا موارد دیگر استفاده کنید. این Layout بسیار شبیه آنچه می‌باشد که ما در tahlilDadeh.com داشتیم.

در اینجا مثالی را از ایجاد یک Layout سه‌سه‌تونی می‌بینیم.

```
<!DOCTYPE html>
<html>
<head>
<title>Three Column HTML Layout</title>
</head>
<body>
<table width="100%" border="0">
<tr valign="top">
<td bgcolor="#aaaa" width="20%">
<b>Main Menu</b><br />
HTML<br />
PHP<br />
PERL...
</td>
<td bgcolor="#b5dcb3" height="200" width="60%">
Technical and Managerial Tutorials
</td>
<td bgcolor="#aaaa" width="20%">
<b>Right Menu</b><br />
HTML<br />
PHP<br />
</td>
</tr>
</table>

```

```
PERL...  
  </td>  
  </tr>  
</table>  
</body>  
</html>
```

این **مثال** نتیجه‌ی زیر را تولید خواهد کرد.

Main Menu	Technical and Managerial Tutorials	Right Menu
HTML		HTML
PHP		PHP
PERL...		PERL...

Span و DIV های HTML – استفاده از Layout

عنصر **<div>** یک عنصر **block level** می باشد که برای گروه بندی عناصر **HTML** استفاده می شود. در حالیکه این عنصر یک عنصر **block level** است، عنصر **** برای گروه بندی عناصر در یک سطح درون خطی استفاده می شوند.

گرچه می توانیم با جدول های **Layout** ، **HTML** های بسیار زیبایی به دست آوریم، اما جدول ها در واقع به عنوان ابزار **Layout** طراحی نشده اند، و بیشتر برای نمایش داده های جدولی استفاده می شوند.

توجه

این **مثال** از **CSS** استفاده می کند، بنابراین قبل از درک این **مثال**، بهتر است درک بهتری از چگونگی کار **CSS** داشته باشید.

مثال

در اینجا سعی می کنیم با استفاده از برچسب **<div>** همان نتیجه ای را به دست اوریم که هنگام استفاده از برچسب **<table>** در مثال قبل به دست آوردیم.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Layouts using DIV, SPAN</title>
</head>
<body>
<div style="width:100%">
<div style="background-color:#b5dcb3; width:100%">
<h1>This is Web Page Main title</h1>
</div>
<div style="background-color:#aaa; height:200px; width:100px; float:left;">
<div><b>Main Menu</b></div>
    HTML<br />
    PHP<br />
    PERL...
</div>
<div style="background-color:#eee; height:200px; width:350px; float:left;">
<p>Technical and Managerial Tutorials</p>
</div>
<div style="background-color:#aaa; height:200px; width:100px; float:right;">
<div><b>Right Menu</b></div>
    HTML<br />
    PHP<br />
    PERL...
</div>
<div style="background-color:#b5dcb3; clear:both">
<center>
    Copyright © 2007 Tahlildadeh.com
</center>
</div>
</div>
</body>
</html>
```

شما می توانید با استفاده از **DIV** و **SPAN** به همراه **Layout, CSS** های بهتری طراحی کنید. برای درک بیشتر از **CSS** لطفا به [CSS Tutorial](#) مراجعه کنید.

آموزش Iframes

شما می توانید یک **frame** درون خطی را با استفاده از برچسب **<iframe>** مربوط به **HTML** تعریف کنید. این برچسب به برچسب **<frameset>** ارتباطی ندارد، در عوض می تواند در هرجایی در داکیومنت شما ظاهر شود.

برچسب `<iframe>` یک محدوده‌ی مستطیلی را در داخل داکیومنت تعریف می‌کند که در آن مرورگر می‌تواند یک داکیومنت مجزا را ارائه دهد، مانند نوارهای اسکرول و حاشیه‌ها.

ویژگی `src` برای مشخص کردن URL مربوط به داکیومنتی استفاده می‌شود که حاوی `frame` درون خطی می‌باشد.

مثال

در زیر مثالی را می‌بینید که چگونگی استفاده از `<iframe>` را توضیح می‌دهد.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Iframes</title>
</head>
<body>
  <p>Document content goes here...</p>
  <iframe src="/html/menu.htm" width="555" height="200">
    Sorry your browser does not support inline frames.
  </iframe>
  <p>Document content also go here...</p>
</body>
</html>
```

ویژگی‌های برچسب `<iframe>`

بسیاری از ویژگی‌های برچسب `<iframe>` شامل نام، گروه، حاشیه، `id`، طول حاشیه، عرض حاشیه، نام، اسکرول کردن، روش و تیتر، درست مانند ویژگی‌های منتظر با برچسب `<frame>` رفتار می‌کنند.

Attribute	Description
<code>src</code>	این ویژگی برای نام گذاری فایلی استفاده می‌شود که باید در <code>frame</code> بارگذاری شود. مقدار آن می‌تواند هر URL باشد. برای مثال <code>src="/html/top_frame.htm"</code> فایل موجود در مسیر <code>html</code> را بارگذاری خواهد کرد.

name	این ویژگی به شما اجازه می دهد تا یک frame را نامگذاری کنید. این ویژگی نشان می دهد که یک داکیومنت در کدام frame باید بارگذاری شود. هنگامی که می خواهید لینک هایی را در یک frame ایجاد کنید که صفحاتی را در یک frame دیگر بارگذاری می کند، که در این مورد دومین frame برای تشخیص خود به عنوان هدف لینک به نام نیاز دارد، در اینجا این ویژگی بسیار مهم است.
frameborder	این ویژگی مشخص می کند که آیا حاشیه های frame نشان داده شوند یا خیر. این ویژگی مقدار داده شده به ویژگی frameborder روی برچسب <frameset> را می گیرد، البته اگر مقدار مشخص شده باشد، این مقدار می تواند 1 (بله) و یا 0 (خیر) باشد.
marginwidth	این ویژگی به شما اجازه می دهد تا عرض فضای بین حاشیه های چپ و راست frame و محتوای آن را مشخص کنید. مقدار به پیکسل داده می شود. برای مثال <code>.marginwidth="10"</code>
marginheight	این ویژگی به شما اجازه می دهد تا طول فضای بین بالا و پایین حاشیه ی frame و محتوای آن را مشخص کنید. مقدار به پیکسل داده می شود. برای مثال <code>marginheight="10"</code>
noresize	به طور پیش فرض هر frame را با کلیک کردن و درگ کردن روی حاشیه های آن می توانید تغییر اندازه دهید. ویژگی noresize مانع تغییر اندازه ی frame توسط یوزر می شود. برای مثال <code>.noresize="noresize"</code>
scrolling	این ویژگی ظاهر نارهای اسکرول نمایان شده در frame را کنترل می کند. این ویژگی مقادیر "auto", "no" یا "yes" را می گیرد. برای مثال <code>scrolling="no"</code> به معنای نبودن نوارهای اسکرول می باشد.

longdesc	<p>این ویژگی به شما اجازه می دهد تا یک لینک به صفحه ای حاوی یک توصیف طولانی از محتوای frame، ارائه بدهید. برای مثال <code>longdesc="framedescription.htm"</code></p>
----------	--

html در background

به طور پیش فرض رنگ زمینه‌ی صفحه‌ی وب شما سفید می‌باشد. ممکن است این زمینه را دوست نداشته باشید، اما نگرانی وجود ندارد. **HTML** دو روش مناسب زیر را ارائه می‌دهد تا زمینه‌ی صفحه‌ی وب خود را به دلخواه بیاراییید.

زمینه‌ی **HTML** با رنگ‌ها.

زمینه‌ی **HTML** با تصاویر.

اکنون اجازه بدهید هر دو روش را یکی با استفاده از **مثال** های مناسب بررسی کنیم.

زمینه‌ی **HTML** با رنگ‌ها

ویژگی **bgcolor** برای کنترل زمینه‌ی یک عنصر **HTML**، به ویژه بدن‌های صفحه و زمینه‌ی جدول، استفاده می‌شود. در زیر ترکیب استفاده از **bgcolor** را با هر برچسب **HTML** می‌بینید.

```
<tagname bgcolor="color_value" ...>
```

این **color-value** می‌تواند به هر کدام از فرمات‌های زیر ارائه شود.

```
<!-- Format 1 - Use color name -->
<table bgcolor="lime">
```

```
<!-- Format 2 - Use hex value -->
<table bgcolor="#f1f1f1">
```

```
<!-- Format 3 - Use color value in RGB terms -->
<table bgcolor="rgb(0,0,120)">
```

مثال

در اینجا **مثال** را می‌بینید از تنظیم زمینه‌ی یک برچسب **HTML**.

```

<!DOCTYPE html>
<html>
<head>
  <title>HTML Background Colors</title>
</head>
<body>
  <!-- Format 1 - Use color name -->
  <table bgcolor="yellow" width="100%">
    <tr>
      <td>
        This background is yellow
      </td>
    </tr>
  </table>
  <!-- Format 2 - Use hex value -->
  <table bgcolor="#6666FF" width="100%">
    <tr>
      <td>
        This background is sky blue
      </td>
    </tr>
  </table>
  <!-- Format 3 - Use color value in RGB terms -->
  <table bgcolor="rgb(255,0,255)" width="100%">
    <tr>
      <td>
        This background is green
      </td>
    </tr>
  </table>
</body>
</html>

```

ویژگی **background** همچنین می تواند برای کنترل زمینه ی یک عنصر **HTML**، به ویژه بدنه ی صفحه و زمینه های جدول، استفاده شود. شما می توانید یک تصویر را به عنوان زمینه ی صفحه و یا جدول **HTML** خود استفاده کنید. در زیر ترکیب استفاده از ویژگی **background** را با هر عنصر **HTML** می بینید.

توجه ویژگی **background** به عنوان ویژگی خوبی تلقی نمی شود و توصیه می شود از **style sheet** برای تنظیم زمینه استفاده کنید.

```
<tagname background="Image URL" ...>
```

متداول ترین فرمت های مورد استفاده ی تصویر عبارتند از **GIF**, **JPEG** و **PNG**.

مثال

در اینجا **مثال** هایی را از تنظیم تصویر به عنوان زمینه ی جدول مشاهده می کنید.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Background Images</title>
</head>
<body>
  <!-- Set table background -->
  <table background="/images/html.jpg" width="100%" height="400">
    <tr>
      <td>
        This background is filled up with HTML image.
      </td>
    </tr>
  </table>
</body>
</html>
```

زمینه های طرح دار و شفاف

شما ممکن است الگوها و زمینه های شفاف بسیاری را در وب سایت های مختلف دیده باشید. این امر به راحتی و با استفاده از تصاویر طرح دار و شفاف در زمینه قابل دستیابی می باشد. توصیه می شود در هنگام ایجاد تصاویر طرح دار یا شفاف **PNG** یا **GIF**، از کوچکترین ابعاد ممکن آنها استفاده کنید، حتی به کوچکی **1x1** برای جلوگیری از بارگذاری آهسته.

مثال

در اینجا **مثال**ی از تنظیم یک الگوی زمینه برای یک جدول را مشاهده می کنید.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Background Images</title>
</head>
<body>
  <!-- Set a table background using pattern -->
  <table background="/images/pattern1.jpg" width="20%" height="100">
    <tr>
      <td>
        This background is filled up with a pattern image.
      </td>
    </tr>
  </table>
  <!-- Another example on table background using pattern -->
  <table background="/images/pattern2.jpeg" width="20%" height="100">
    <tr>
      <td>
        This background is filled up with a pattern image.
      </td>
    </tr>
  </table>
```

```
</td>
</tr>
</table>
</body>
</html>
```

رنگهای HTML

رنگ ها برای دادن یک ظاهر و احساس خوب به وب سایت شما بسیار مهم می باشند. شما می توانید رنگ ها را روی لایه های صفحه با استفاده از برچسب **<body>** مشخص کنید، یا می توانید رنگ ها را برای برچسب های مجزا با استفاده از ویژگی **bgcolor** مشخص کنید.

برچسب **<body>** دارای ویژگی های زیر می باشد که می تواند برای تنظیم رنگ های مختلف استفاده شود.

رنگی را برای زمینه صفحه تنظیم می کند.

رنگی را برای متن تنظیم می کند.

رنگی را برای لینک های فعال یا انتخاب شده تنظیم می کند.

رنگی را برای لینک شده تنظیم می کند.

رنگی را برای لینک های بازدید شده تنظیم می کند. یعنی برای متن لینک شده که روی آن کلیک کرده اید.

روش های کدگذاری رنگ

در زیر سه روش متفاوت برای تنظیم رنگ در صفحه و ب خود مشاهده می کنید

– می توانید نام رنگ ها را به طور مستقیم تعیین کنید، به عنوان **مثال** سبز، قرمز، آبی و غیره.

– یک کد شش رقمی که نشان دهنده مقدار آبی، قرمز و سبز سازنده رنگ، می باشد.

– این مقدار با استفاده از ویژگی **rgb()** تعیین می شود.

اکنون این روش های رنگ گذاری را یکی یکی بررسی خواهیم کرد.

رنگ های HTML – نام رنگ

می توانید برای تنظیم رنگ متن یا زمینه به طور مستقیم از نام رنگ استفاده کنید. **W3C** لیستی از 16 رنگ پایه را دارد که توسط یک اعتبار سنج **HTML** ارزیابی می شود، اما نام بیشتر از 200 رنگ مختلف وجود دارد که توسط مرورگرهای مهم پشتیبانی می شوند.

16 رنگ استاندارد W3C

در اینجا نام 16 رنگ استاندارد **W3C** را مشاهده می کنید، پیشنهاد می شود که از آنها استفاده کنید.

Black	Gray	Silver	White
Yellow	Lime	Aqua	Fuchsia
Red	Green	Blue	Purple
Maroon	Olive	Navy	Teal

مثال

در اینجا **مثال** هایی را میبینید از تنظیم زمینه ی یک برچسب **HTML** به وسیله ی نام رنگ.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Colors by Name</title>
</head>
<body text="blue" bgcolor="green">
  <p>Use different color names for body and table and see the result.</p>
  <table bgcolor="black">
    <tr>
      <td>
        <font color="white">This text will appear white on black background.</font>
      </td>
    </tr>
  </table>
</body>
</html>
```

رنگ های HTML – کدهای Hex

هگزادسیمال یک نمایش 6 رقمی از یک رنگ می باشد. دو رقم اول یعنی RR نمایانگر رنگ قرمز (Red) می باشند، دو رقم بعدی GG مقدار سبز (green) را نشان می دهند و آخرین دو رقم BB نیز مقدار آبی (blue) را نشان می دهند.

هر کد هگزادسیمال به وسیله ی یک علامت # دنبال می شود. در ادامه رنگ های مورد استفاده در نشانه گذاری هگزادسیمال را مشاهده می کنید.

Color	Color HEX
	#000000
	#FF0000
	#00FF00
	#0000FF
	#FFFF00
	#00FFFF
	#FF00FF
	#C0C0C0
	#FFFFFF

در زیر مثال هایی را می بینند از تنظیم زمینه‌ی یک برچسب HTML به وسیله‌ی کد رنگ در هگزادسیمال.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Colors by Hex</title>
</head>
<body text="#0000FF" bgcolor="#00FF00">
<p>Use different color hexa for body and table and see the result.</p>
<table bgcolor="#000000">
  <tr>
    <td>
      <font color="#FFFFFF">This text will appear white on black background.</font>
    </td>
  </tr>
</table>
</body>
</html>
```

رنگ‌های HTML – مقادیر RGB

مقدار این رنگ با استفاده از ویژگی (rgb()) مشخص می‌شود. این ویژگی سه مقدار می‌گیرد که هر کدام برای سبز، قرمز و آبی می‌باشد. مقدار می‌تواند عددی بین 0 و 255 باشد.

توجه همه‌ی مرورگرها ویژگی (rgb()) را پشتیبانی نمی‌کنند، بنابراین توصیه می‌شود از آن استفاده نکنید.

در زیر لیستی از رنگ‌ها با مقادیر RGB را مشاهده می‌کنید.

Color	Color RGB
	rgb(0,0,0)
	rgb(255,0,0)
	rgb(0,255,0)

	rgb(0,0,255)
	rgb(255,255,0)
	rgb(0,255,255)
	rgb(255,0,255)
	rgb(192,192,192)
	rgb(255,255,255)

در اینجا مثال هایی را مشاهده می کنید از تنظیم زمینه‌ی یک برچسب HTML به وسیله‌ی برچسب `rgb()` با کد رنگ.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Colors by RGB code</title>
</head>
<body text="rgb(0,0,255)" bgcolor="rgb(0,255,0)">
  <p>Use different color code for body and table and see the result.</p>
  <table bgcolor="rgb(0,0,0)">
    <tr>
      <td>
        <font color="rgb(255,255,255)">This text will appear white on black background.</font>
      </td>
    </tr>
  </table>
</body>
</html>>
```

در زیر لیستی از 216 رنگ را مشاهده می کنید که تصور می شود سالم ترین و مستقل ترین رنگ های کامپیوتر باشند. این رنگ ها از کد هگزای 000000 تا FFFFFF متفاوت می باشند و توسط همه می کامپیوترهای دارای پالت رنگ 256 تابی پشتیبانی می شوند.

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

666600	666633	666666	666699	6666CC	6666FF
669900	669933	669966	669999	6699CC	6699FF
66CC00	66CC33	66CC66	66CC99	66CCCC	66CCFF
66FF00	66FF33	66FF66	66FF99	66FFCC	66FFFF
990000	990033	990066	990099	9900CC	9900FF
993300	993333	993366	993399	9933CC	9933FF
996600	996633	996666	996699	9966CC	9966FF
999900	999933	999966	999999	9999CC	9999FF
99CC00	99CC33	99CC66	99CC99	99CCCC	99CCFF
99FF00	99FF33	99FF66	99FF99	99FFCC	99FFFF
CC0000	CC0033	CC0066	CC0099	CC00CC	CC00FF
CC3300	CC3333	CC3366	CC3399	CC33CC	CC33FF
CC6600	CC6633	CC6666	CC6699	CC66CC	CC66FF
CC9900	CC9933	CC9966	CC9999	CC99CC	CC99FF

CCCC00	CCCC33	CCCC66	CCCC99	CCCCCC	CCCCFF
CCFF00	CCFF33	CCFF66	CCFF99	CCFFCC	CCFFFF
FF0000	FF0033	FF0066	FF0099	FF00CC	FF00FF
FF3300	FF3333	FF3366	FF3399	FF33CC	FF33FF
FF6600	FF6633	FF6666	FF6699	FF66CC	FF66FF
FF9900	FF9933	FF9966	FF9999	FF99CC	FF99FF
FFCC00	FFCC33	FFCC66	FFCC99	FFCCCC	FFCCFF
FFFF00	FFFF33	FFFF66	FFFF99	FFFFCC	FFFFFF

آموزشگاه حلبچهاردهه

فونت ها

فونت ها نقش مهمی در زیبای ساختن و خوانانتر کردن وب سایت بازی می کنند. ظاهر و رنگ فونت به طور کامل بستگی به کامپیوتر و مرورگر دارد که استفاده می شود، اما شما می توانید از برچسب `` در HTML برای افزودن استایل، سایز و رنگ به متن خود در وب سایت استفاده کنید. می توانید از یک برچسب `<basefont>` برای تنظیم تمام متن خود به اندازه، ظاهر و رنگ یکسان استفاده کنید.

برچسب فونت دارای سه ویژگی به نام های `color` و `size` و `face` می باشد که فونت شکا را به دلخواه در می آورد. برای تغییر هر کدام از ویژگی های فونت در هر زمانی در صفحه ی وب خود، به سادگی از برچسب `` استفاده کنید. متنی که دنبال می کند، تغییر یافته با قی می ماند تا زمانی که شما آن را با `` برچسب بیندید. شما می توانید یکی از ویژگی ها یا همه ی ویژگی های داخل برچسب `` را تغییر دهید.

نکته

برچسب های **font** و **basefont** استفاده نمی شوند و احتمال می رود که در ورژن های بعدی **HTML** حذف شوند. بنابراین نباید مورد استفاده قرار بگیرند، پیشنهاد می شود که برای اجرای فونت های خود از استایل های **CSS** استفاده کنید. اما برای رسیدن به هدف این فصل برچسب های **basefont** و **font** را جزئیات توضیح می دهد.

تنظیم اندازه فونت

شما می توانید با استفاده از ویژگی **size** اندازه فونت محتوا را تنظیم کنید. دامنه **ی مقادیر قابل قبول از 1 (کوچکترین) تا 7 (بزرگترین)** می باشد. اندازه فونت پیش فرض 3 می باشد.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Setting Font Size</title>
</head>
<body>
  <font size="1">Font size="1"</font><br />
  <font size="2">Font size="2"</font><br />
  <font size="3">Font size="3"</font><br />
  <font size="4">Font size="4"</font><br />
  <font size="5">Font size="5"</font><br />
  <font size="6">Font size="6"</font><br />
  <font size="7">Font size="7"</font>
</body>
</html>
```

اندازه فونت مربوط

شما می توانید مشخص کنید چه تعداد از فونت ها بزرگتر و چه تعداد کوچکتر از اندازه فونت حاضر باشند. می توانید آن را مانند **** or **** مشخص کنید.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Relative Font Size</title>
</head>
<body>
  <font size="-1">Font size="-1"</font><br />
  <font size="+1">Font size="+1"</font><br />
```

```

<font size="+2">Font size="+2"</font><br />
<font size="+3">Font size="+3"</font><br />
<font size="+4">Font size="+4"</font>
</body>
</html>

```

تنظیم ظاهر فونت

شما می توانید با استفاده از ویژگی **face** ظاهر فونت را تنظیم کنید، اما باید بدانید که اگر کاربر بازدیدکنندهٔ صفحه، فونت را نصب نکرده باشد، قادر به دیدن آن نخواهد بود. در عوض کاربر ظاهر فونت پیش فرض را می بیند که برای کامپیوترش مناسب می باشد.

```

<!DOCTYPE html>
<html>
<head>
  <title>Font Face</title>
</head>
<body>
  <font face="Times New Roman" size="5">Times New Roman</font><br />
  <font face="Verdana" size="5">Verdana</font><br />
  <font face="Comic sans MS" size="5">Comic Sans MS</font><br />
  <font face="WildWest" size="5">WildWest</font><br />
  <font face="Bedrock" size="5">Bedrock</font><br />
</body>
</html>

```

مثال

تعیین ظاهر فونت جایگزین

یک بازدید کننده فقط قادر خواهد بود فونت شما را ببیند، اگر آن را نصب شده روی کامپیوتر خود داشته باشد. بنابراین امکان تعیین دو یا بیشتر از دو ظاهر جایگزین با ارائهٔ نام های مربوط به فونت‌ها وجود دارد.

```

<font face="arial,helvetica">
<font face="Lucida Calligraphy,Comic Sans MS,Lucida Console">

```

اگر هیچکدام از فونت‌های ارائه شده نصب نشده باشند، بنابراین فونت پیش فرض *Times New Roman* نمایش داده خواهد شد.

تنظیم رنگ فونت

شما می توانید با استفاده از ویژگی `color` هر رنگی را برای فونت تنظیم کنید. شما می توانید رنگ مورد نظر خود را یا با استفاده از نام رنگ و یا با استفاده از کد هگزادسیمال برای آن رنگ تعیین کنید.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Setting Font Color</title>
</head>
<body>
  <font color="#FF00FF">This text is in pink</font><br />
  <font color="red">This text is red</font>
</body>
</html>
```

عنصر `<basefont>`

انتظار می رود که عنصر `<basefont>` یک اندازه، رنگ و ظاهر پیش فرض برای هر بخشی از داکیومنت که در برچسب `` قرار نمی گیرند، تنظیم کند. شما می توانید از عناصر `` استفاده کنید تا تنظیمات `<basefont>` را انجام دهید.

برچسب `<basefont>` همچنین ویژگی های رنگ، اندازه و ظاهر را می گیرد و با دادن مقدار بیشتر از `+1` برای فونت های بزرگتر و کمتر از `-2` برای فونت های کوچکتر، تنظیمات فونت مربوطه را پشتیبانی می کند.

مثال

```
<!DOCTYPE html>
<html>
<head>
  <title>Setting Basefont Color</title>
</head>
<body>
  <basefont face="arial, verdana, sans-serif" size="2" color="#ff0000">
    <p>This is the page's default font.</p>
    <h2>Example of the <basefont> Element</h2>
    <p>
      <font size="+2" color="darkgray">
        This is darkgray text with two sizes larger
      </font>
    </p>
    <p>
    </p>
  </basefont>
</body>
</html>
```

```

<font face="courier" size="-1" color="#000000">
It is a courier font, a size smaller and black in color.
</font>
</p>
</body>
</html>

```

frm های html

وقتی که می خواهید داده هایی را از سایت بازدیدکننده جمع آوری کنید، به فرم های HTML احتیاج خواهید داشت. به عنوان مثال در هنگام ثبت کاربر، اطلاعاتی مانند نام، آدرس ایمیل و کارت اعتباری و غیره را جمع آوری می کنید. یک فرم داده ها را از بازدیدکننده سایت می گیرد و سپس آن را به یک برنامه پایانی مانند ASP، اسکریپت PHP یا اسکریپت CGI تعریف شده در برنامه، روی داده ی منتقل شده انجام می دهد.

عناصر متفاوتی برای فرم در دسترس میباشند، مانند فیلد های متن، فیلد های textarea، منوهای رو به پایین، دکمه های رادیو، چک باکس ها و غیره.

برچسب <form> مربوط به HTML، برای ایجاد یک فرم استفاده می شود و دارای ترکیب زیر می باشد.

```

<form action="Script URL" method="GET|POST">
  form elements like input, textarea etc.
</form>

```

ویژگی های فرم

علاوه بر ویژگی های متداول، در زیر لیستی از متداول ترین ویژگی های مربوط به فرم را مشاهده می کنید.

Attribute	Description
action	اسکریپت backend برای پردازش داده ی انتقال شده ی شما آماده میباشد.

method	روشی برای بارگذاری داده. متد اولترين روش های مورد استفاده GET و POST می باشند.
target	پنجره‌ی هدف یا چارچوب را، جایی که نتیجه‌ی اسکریپت نمایش داده خواهد شد، مشخص می‌کند. این ویژگی مقادیر _blank, _self, _parent و غیره را به خود می‌گیرد.
enctype	<p>You can use the enctype attribute to specify how the browser encodes the data before it sends it to the server. Possible values are</p> <p>شما می‌توانید از ویژگی enctype برای مشخص کردن چگونگی برنامه نویسی توسط مرورگر قبل از ارسال به سرور، استفاده کنید. مقادیر ممکن عبارتند از</p> <ul style="list-style-type: none"> – application/x-www-form-urlencoded بیشتر فرم‌ها در سناریو‌های ساده استفاده می‌کنند. – multipart/form-data داده‌های دوتایی را به شکل فایل‌هایی مانند تصویر بارگذاری کنید.

کنترل‌های فرم HTML

انواع مختلفی از کنترل‌های فرم وجود دارند که می‌توانید برای جمع آوری داده با استفاده از فرم **HTML** از آنها استفاده کنید.

کنترل‌های ورودی متن

کنترل‌های چک باکس

کنترل های رادیو باکس

کنترل های انتخاب باکس

باکس های انتخاب فایل

کنترل های مخفی

دکمه های قابل کلیک شدن

ثبت و تنظیم مجدد

کنترل های ورودی متن

سه نوع ورودی متن وجود دارد که در فرم ها استفاده می شوند .

کنترل های ورودی متن تک خطی این کنترل برای آیتم هایی استفاده می شود که فقط یک خط از ورودی کاربر را لازم دارد، مانند باکس های جستجو و نام ها. این کنترل ها با استفاده از برچسب `<input>` مریبوط به `HTML` ایجاد می شوند.

کنترل ورودی گذرواژه این نیز یک متن ورودی تک خطی می باشد، اما به محض اینکه کاربر کاراکتر را وارد می کند، این کنترل آن را می پوشاند.

کنترل های متن ورودی چند خطی زمانی استفاده می شود که لازم است کاربر جزئیاتی را وارد کند که بیشتر از یک جمله می باشند. کنترل های ورودی چند خطی با استفاده از برچسب `<textarea>` استفاده می شوند.

کنترل های ورودی تک خطی

این کنترل ها برای آیتم هایی استفاده می شوند که کاربر فقط یک خط ورودی احتیاج دارد، مانند باکس های جستجو و نام ها و با استفاده از برچسب `<input>` مریبوط به `HTML` ایجاد می شوند.

مثال

در اینجا مثال پایه‌ای می‌بینید از یک ورودی تک خطی که برای گرفتن نام کوچک و نام خانوادگی به کار می‌رود.

```
<!DOCTYPE html>
<html>
<head>
<title>Text Input Control</title>
</head>
<body>
<form>
First name <input type="text" name="first_name" />
<br>
Last name <input type="text" name="last_name" />
</form>
</body>
</html>
```

در زیر لیستی از ویژگی‌های برچسب **<input>** را برای ایجاد فیلد متن می‌بینید.

Attribute	Description
type	نوع کنترل ورودی را نشان می‌دهد و برای کنترل متن ورودی برای text تنظیم خواهد شد.
name	برای دادن نام به کنترل استفاده می‌شود که قرار است برای تشخیص به سرور ارسال شود و مقدار بگیرد.
value	می‌تواند برای ارائه‌ی یک مقدار اصلی در داخل کنترل استفاده شود.
size	اجازه می‌دهد تا عرض کنترل متن ورودی را متناسب با کاراکترها تعیین کنید.
maxlength	اجازه می‌دهد تا حداقل تعداد کاراکترهایی را که یک کاربر می‌تواند در یک تکست باکس وارد کند، مشخص کنید.

کنترل های پسورد ورودی

این کنترل نیز یک کنترل تک خطی می باشد، اما به محض اینکه کاربر کاراکترها را وارد می کند، آنها را می پوشاند. این ها نیزیا استفاده از برچسب **HTML** **<input>** مریبوط به ایجاد می شوند، اما نوع ویژگی با عنوان **password** تنظیم میشود.

```
<!DOCTYPE html>
<html>
<head>
  <title>Password Input Control</title>
</head>
<body>
<form>
  User ID <input type="text" name="user_id" />
  <br>
  Password <input type="password" name="password" />
</form>
</body>
</html>
```

مثال

در اینجا مثالی از ورودی تک خطی پسورد می بینید که برای گرفتن پسورد کاربر استفاده می شود.

ویژگی ها

در زیر لیست مریبوط به ویژگی های برچسب **<input>** را برای ایجاد فیلد پسورد می بینید.

Attribute	Description
type	نوع کنترل ورودی را نشان می دهد و برای کنترل ورودی پسورد، برای password تنظیم خواهد شد.
name	برای نام گذاری کنترلی استفاده می شود که برای تشخیص و گرفتن مقدار به سرور فرستاده می شود.

value	برای ارائه‌ی یک مقدار اولیه در داخل کنترل استفاده می‌شود.
size	اجازه می‌دهد تا عرض کنترل متن ورودی را با کاراکترها تعیین کنید.
maxlength	اجازه می‌دهد تا حداقل تعداد کاراکترهایی را که یک کاربر می‌تواند در یک تکست باکس وارد کند، تعیین کنید.

کنترل‌های متن ورودی چند خطی

زمانی استفاده می‌شود که یک کاربر باید جزئیاتی را ظارد کند که بیشتر از یک جمله می‌باشد. کنترل‌های

ورودی چند خطی با استفاده از برچسب `<textarea>` ایجاد می‌شوند.

مثال

در اینجا مثالی را می‌بینید از یک ورودی متن چند خطی که برای ارائه‌ی توصیفات آیتم استفاده می‌شود.

```
<!DOCTYPE html>
<html>
<head>
  <title>Multiple-Line Input Control</title>
</head>
<body>
<form>
  Description <br />
  <textarea rows="5" cols="50" name="description">
    Enter description here...
  </textarea>
</form>
</body>
</html>
```

ویژگی‌ها

در زیر لیستی از ویژگی‌های برچسب `<textarea>` ارائه شده‌اند.

Attribute	Description
name	برای نامگذاری کنترلی استفاده می شود که به سرور ارسال می شود تا تشخیص داده شده و مقدار بگیرد.
rows	تعداد ردیف های area box مربوط به متن را نشان می دهد.
cols	تعداد ستون های area box مربوط به متن را نشان می دهد.

کنترل چک باکس

چک باکس ها زمانی استفاده می شوند که بیشتر از یک گزینه قرار است انتخاب شود. آنها نیز با استفاده از برچسب `<input>` ایجاد می شوند، اما نوع ویژگی به `checkbox` تنظیم می شود.

مثال

در اینجا مثالی از کد `HTML` را مشاهده می کنید برای یک فرم با دو چک باکس.

```
<!DOCTYPE html>
<html>
<head>
  <title>Checkbox Control</title>
</head>
<body>
  <form>
    <input type="checkbox" name="maths" value="on"> Maths
    <input type="checkbox" name="physics" value="on"> Physics
  </form>
</body>
</html>
```

در زیر لیستی از ویژگی های برچسب `<checkbox>` را می بینید

Attribute	Description
type	نوع کنترل ورودی را نشان می دهد و برای کنترل ورودی چک باکس با checkbox کنترل خواهد شد.
name	برای نامگذاری کنترلی استفاده می شود که برای تشخیص و گرفتن مقدار به سرور ارسال می شود.
value	مقداری که اگر چک باکس انتخاب شود، استفاده خواهد شد.
checked	اگر بخواهید آن را به طور پیش فرض انتخاب کنید، با checked تنظیم می شود.

کنترل دکمه‌ی رادیو

دکمه های رادیو زمانی استفاده میشوند که بین چندین گزینه تنها یک گزینه باید انتخاب شود. این کنترل ها نیز با برچسب **<input type="radio"** ایجاد میشوند، اما نوع ویژگی با **radio** تنظیم می شود.

مثال

در اینجا مثالی از کد **HTML** می بینید برای یک فرم با دو دکمه‌ی رادیو.

```
<!DOCTYPE html>
<html>
<head>
  <title>Checkbox Control</title>
</head>
<body>
  <form>
    <input type="radio" name="maths" value="on"> Maths
    <input type="radio" name="physics" value="on"> Physics
  </form>
</body>
</html>
```

ویژگی ها

در زیر لیستی از ویژگی های دکمه‌ی رادیو را می بینید.

Attribute	Description
type	نوع کنترل ورودی را نشان می دهد و برای کنترل ورودی چک باکس با عنوان radio تنظیم می شود.
name	برای نامگذاری کنترلی استفاده می شود که برای تشخیص و گرفتن مقدار به سرور ارسال می شود.
value	مقداری که radio box انتخاب شود، استفاده خواهد شد.
checked	اگر می خواهید آن را به عنوان پیش فرض استفاده کنید، به checked تنظیم کنید.

کنترل Select Box

یک منوی رو به پایین نیز نامیده می شود، گزینه ای را برای ارائه ی گزینه های مختلف به شکل یک لیست رو به پایین ارائه می دهد، که کاربر می تواند از آن یک یا بیشتر از یک گزینه را انتخاب کند.

مثال

در اینجا مثالی از کد HTML برای یک فرم با یک جعبه ای رو به پایین می بینید.

```
<!DOCTYPE html>
<html>
<head>
  <title>Select Box Control</title>
</head>
<body>
<form>
<select name="dropdown">
  <option value="Maths" selected>Maths</option>
  <option value="Physics">Physics</option>
</select>
```

```
</form>  
</body>  
</html>
```

در زیر لیستی از ویژگی های برچسب **<select>** را می بینید.

Attribute	Description
name	برای نامگذاری کنترلی استفاده می شود که برای تشخیص و گرفتن مقدار به سرور انتقال می شود.
size	برای نمایش یک لیست باکس اسکرولینگ استفاده می شود.
multiple	اگر روی multiple تنظیم شده باشد، به کاربر اجازه می دهد تا چند آیتم را از منو انتخاب کند.

در زیر لیستی از ویژگی های مهم برچسب **<option>** ارائه شده است.

Attribute	Description
value	اگر در سلکت باکس یک گزینه انتخاب شده باشد، این مقدار استفاده خواهد شد.
selected	مشخص می کند که این گزینه رد زمان بارگذاری صفحه، باید گزینه‌ی انتخابی باشد.
label	یک روش جایگزین برای برچسب دار کردن گزینه ها.

فایل آپلود باکس

اگر می خواهید به کاربر اجازه دهید تا فایلی را روی وب سایت شما آپلود کند، به یک کنترل فایل آپلود احتیاج خواهید داشت، که **Select Box** نیز نامیده می شود. این ویژگی با استفاده از برچسب **<input>** نیز ایجاد می شود، اما نوع ویژگی با **file** تنظیم می شود.

مثال

در اینجا مثالی را می بینید از کد **HTML** برای یک فرم با یک فایل آپلود باکس.

```
<!DOCTYPE html>
<html>
<head>
  <title>File Upload Box</title>
</head>
<body>
  <form>
    <input type="file" name="fileupload" accept="image/*" />
  </form>
</body>
</html>
```

ویژگی ها

در زیر لیستی از ویژگی های فایل آپلود باکس را می تبلیغ.

Attribute	Description
name	برای نامگذاری کنترلی استفاده می شود که برای تشخیص و گرفتن مقدار به سرور انتقال می شود.
accept	نوع فایل هایی را که سرور می پذیرد، تعیین می کند.

کنترل های دکمه

راه های مختلفی برای ایجاد دکمه های قابل کلیک شدن وجود دارد. شما با استفاده از برقسپ `<input>` و با تنظیم نوع ویژگی آن به `button` دکمه های قابل کلیک شدن ایجاد کنید. نوع ویژگی می تواند مقادیر زیر را بگیرد.

Type	Description
submit	دکمه ای را ایجاد می کند که به طور خودکار یک فرم را می پذیرد.
reset	دکمه ای را ایجاد می کند که به طور خودکار کنترل های فرم را به مقادیر اولیه خود بازمی گرداند.
button	دکمه ای را ایجاد می کند که برای اجرای اسکریپت کاربر، زمانی که کاربر آن دکمه را کلیک می کند، استفاده می شود.
image	یک دکمه ای قابل کلیک شدن ایجاد می گند، اما ما می توانیم از یک تصویر به عنوان زمینه ی دکمه استفاده کنیم.

مثال

در اینجا مثالی از کد `HTML` برای یک فرم با سه نوع دکمه را می بینید.

```
<!DOCTYPE html>
<html>
<head>
  <title>File Upload Box</title>
</head>
<body>
  <form>
    <input type="submit" name="submit" value="Submit" />
    <input type="reset" name="reset" value="Reset" />
    <input type="button" name="ok" value="OK" />
  </form>
</body>
</html>
```

```
<input type="image" name="imagebutton" src="/html/images/logo.png" />
</form>
</body>
</html>
```

کنترل های مخفی شده ی فرم

کنترل های مخفی شده فرم برای پنهان کردن داده در داخل صفحه ای استفاده می شوند که بعدا می توانند به سرور ارسال شوند. این کنترل در داخل کد مخفی شده و روی صفحه ی حقیقی ظاهر نمی شود. برای مثال فرم مخفی شده ی زیر برای حفظ شماره ی صفحه ی جاری استفاده می شود. وقتی که کاربر روی **next page** کلیک کند، مقدار کنترل مخفی شده به سرور وب ارسال شده و در آنجا تصمیم خواهد گرفت که بر اساس انتقال صفحه ی جاری، کدام صفحه نمایش داده خواهد شد.



```
<!DOCTYPE html>
<html>
<head>
  <title>File Upload Box</title>
</head>
<body>
  <form>
    <p>This is page 10</p>
    <input type="hidden" name="pagename" value="10" />
    <input type="submit" name="submit" value="Submit" />
    <input type="reset" name="reset" value="Reset" />
  </form>
</body>
</html>
```

آموزش چند رسانه ای در HTML

گاهی اوقات تمایل دارید که موسیقی یا ویدئو به صفحه ی وب خود اضافه کنید. ساده ترین راه برای افزودن صدا یا ویدئو به وب سایت، وارد کردن برچسب خاص **<embed>** به نام **HTML** می باشد. این برچسب باعث می شود که مرورگر به طور خودکار کنترل هایی را برای مولتی مدیا وارد کند که مرورگر ارائه شده برچسب **<embed>** و نوع مديای ارائه شده را پشتیبانی می کند.

همچنین می توانید یک برچسب `<noembed>` برای مروگرهای وارد کنید که برچسب `<embed>` را نمی شناسند. به عنوان **مثال** می توانید از `<embed>` برای نمایش یک فیلم به انتخاب خود استفاده کنید و اگر مروگر `<embed>` را پشتیبانی نمی کند، از

برای نمایش یک تصویر **JPG** مجزا استفاده کنید.



```
<!DOCTYPE html>
<html>
<head>
  <title>HTML embed Tag </title>
<head>
<body>
  <embed src="/html/yourfile.mid" width="100%" height="60">
  <noembed></noembed>
  <embed>
<body>
<html>
```

در اینجا **مثال** ساده ای می بینید از اجرای یک فایل جاسازی شده **midi**.

ویژگی های برچسب `<embed>`

در زیر لیستی از ویژگی های مهمی را می بینید که توسط برچسب `<embed>` استفاده می شود.

Attribute	Description
align	تعیین می کند که چگونه یک آبجکت را تنظیم کرد، که می تواند هم در مرکز، هم در راست و یا چپ تنظیم شود.
autoplay	این ویژگی Boolean نشان می دهد که آیا میدیا باید به طور خودکار شروع به کار کند. شما می توانید آن را با <code>true</code> یا <code>false</code> تنظیم کنید.

loop	تعیین می کند که آیا صدا متدالو پشت سرهم تکرار شود (لوب را روی true تنظیم کنید)، یا چند دفعه باید تکرار شود (یک مقدار مثبت) و یا اصلا تکرار نشود(false).
playcount	تعداد دفعاتی را که یک صدا باید تکرار شود تعیین می کند. اگر در حال استفاده از IE هستید، این گزینه جایگزینی برای loop می باشد.
hidden	مشخص می کند که آیا آبجکت مولتی مدیا باید روی صفحه نمایش داده شود. مقدار false یعنی نه و مقدار true یعنی بله.
width	عرض آبجکت به پیکسل.
height	عرض آبجکت به پیکسل.
name	نامی که برای اشاره به آبجکت استفاده می شود.
src	URL مربوط به آبجکت که باید جاسازی شود.
volume	میزان صدا را کنترل می کند که می تواند از 0 (صدا قطع شده) تا 100 (آخرین حد صدا) باشد.

انواع ویدیوهای پشتیبانی شده

می توانید از انواع مدیاهای مختلفی مانند فایل های **AVI**, **MOV** و **flash movies** در داخل برچسب **embed** استفاده کنید.

فایل های **swf** – فایل هایی هستند که با برنامه **macromedia's flash** تولید می شوند.

فایل های **wmv** – انواع فایل های تصویری ویندوز مایکروسافت می باشند.

فایل های Quick time movie mov – فرمت در اپل.

فایل های Moving – فایل های تصویری هستند که توسط گروه تخصصی تصاویر متحرک (Moving Pictures Expert Group) ایجاد می شوند.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML embed Tag</title>
</head>
<body>
<embed src="/html/yourfile.swf" width="200" height="200">
<noembed></noembed>
</embed>
</body>
</html>
```

صدا زمینه

شما می توانید از برچسب **html** برای اجرای صدا در زمینه ی صفحه ی وب خود استفاده کنید. این برچسب فقط توسط اینترنت اکسپلورر پشتیبانی می شود و دیگر مرورگرها آن را نادیده می گیرند. زمانی که سند اصلی در ابتدا توسط کاربر دانلود شده و نمایش داده می شود، این برچسب یک فایل صدا را دانلود کرده و اجرا می کند. همچنین هر وقت کاربر مرورگر را تازه سازی کند، صدا زمینه نیز دوباره اجرا خواهد شد.

این برچسب دارای فقط دو ویژگی می باشد، **src** و **loop**، که همانطور که در بالا توضیح داده شد هر دوی این ویژگی ها دارای یک معنا می باشند.

در اینجا **مثال** ساده ای از اجرای فایل کوچک midi را می بینید.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML embed Tag</title>
</head>
<body>
<bgsound src="/html/yourfile.mid">
<noembed></noembed>
</bgsound>
</body>
</html>
```

این مثال یک صفحه‌ی خالی تولید خواهد کرد. این برچسب هیچ مولفه‌ای را اجرا نمی‌کند و مخفی باقی می‌ماند.

اینترنت اکسپلورر نیز تنها سه فرمت صدا را اجرا می‌کند **wav** فرمت داخلی کامپیوترها، **au** فرمت داخلی برای کار ابزارهای **Unix** و **MIDI** یک کد برنامه‌گذاری جهانی برای موسیقی.

برچسب آبجکت در HTML

HTML 4 عنصر **<object>** را معرفی می‌کند که یک راه حل چند منظوره برای وارد کردن آبجکت عمومی ارائه می‌دهد. عنصر **<object>** به نویسندهای **HTML** اجازه می‌دهد تا هر چیز لازم را با یک آبجکت برای ارائه‌ی آن توسط یک کاربر، تعیین کند.

در اینجا چند مثال در این رابطه می‌بینید.

مثال 1

شما می‌توانید یک سند **HTML** را در خود سند **HTML** اجرا کنید.

```
<object data="data/test.htm" type="text/html" width="300" height="200">
  alt <a href="data/test.htm">test.htm<a>
</object>
```

در اینجا اگر مرورگر برچسب **object** را پشتیبانی نکند، ویژگی **alt** وارد تصویر می‌شود.

مثال 2

شما می‌توانید یک سند **PDF** را در یک سند **HTML** اجرا کنید.

```
<object data="data/test.pdf" type="application/pdf" width="300" height="200">
  alt <a href="data/test.pdf">test.htm<a>
</object>
```

مثال 3

شما می توانید با استفاده از برجسب **<param>** برخی پارامترهای متناسب با سند را مشخص کنید. در اینجا مثالی از اجرای فایل wav را می بینید.

```
<object data="data/test.wav" type="audio/x-wav" width="200" height="20">
<param name="src" value="data/test.wav">
<param name="autoplay" value="false">
<param name="autoStart" value="0">
alt <a href="data/test.wav">test.wav<a>
</object>
```

مثال 4

شما می توانید یک سند flash مانند زیر اجرا کنید.

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" id="penguin"
       codebase="someplace/swflash.cab" width="200" height="300">
<param name="movie" value="flash/penguin.swf" />
<param name="quality" value="high" />

</object>
```

مثال 5

شما می توانید یک سند java applet وارد سند HTML کنید.

```
<object classid="clsid:8ad9c840-044e-11d1-b3e9-00805f499d93"
       width="200" height="200">
<param name="code" value="applet.class">
</object>
```

ویژگی **classid** تعیین می کند که چه نسخه ای از java plug-in استفاده شود. شما می توانید از ویژگی انتخابی **codebase** برای مشخص کردن چگونگی دانلود JRE استفاده کنید.

آموزش HTML Marquee

یک قطعه اسکرولینگ می باشد که یا به صورت افقی در عرض متن و یا به صورت عمودی در پایین صفحه ی وب شما نمایش داده می شود، بستگی به تنظیمات دارد و توسط برجسب **<Fmarquees>** ایجاد می شود.

توجه ممکن است برچسب **<marquees>** در **HTML** توسط مرورگرهای زیادی پشتیبانی نشود، بنابراین توصیه می شود که به این برچسب تکیه نکنید، در عوض می توانید از **CSS** و **javascript** برای ایجاد چنین تاثیراتی استفاده کنید.

ترکیب ساده برای استفاده از برچسب **<marquees>** در **HTML** مانند زیر می باشد

```
<marquee attribute_name="attribute_value" ....more attributes>
One or more lines or text message or image
</marquee>
```

ویژگی های برچسب <marquees>

در زیر لیست مهمی از ویژگی های برچسب **<marquees>** را می توانید مشاهده کنید.

Attribute	Description
width	این ویژگی عرض marquee را مشخص می کند. می تواند مقداری مانند 10 یا 20 درصد باشد.
height	این ویژگی طول marquee را مشخص می کند. می تواند مقداری مانند 10 یا 20 درصد باشد.
direction	این ویژگی مسیری را که marquee باید در آن اسکرول شود، تعیین می کند. می تواند مقداری مانند up, down, left و right داشته باشد.
behavior	این ویژگی نوع اسکرول marquee را تعیین می کند. می تواند مقداری مانند scroll, alternate و side داشته باشد.
scrolldelay	این ویژگی میزان تاخیر بین دو پرش را تعیین می کند. می تواند مقداری مانند 10 داشته باشد.
scrollamount	سرعت متن marquee را تعیین می کند. می تواند مقداری مانند 10 داشته باشد.

loop	تعداد دفعات loop را تعیین می کند. مقدار پیش فرض INFINITE می باشد که به این معناست که در <code>marquee</code> به طور پایان ناپذیری loop انجام می شود.
bgcolor	این ویژگی رنگ زمینه را به شکل نام رنگ یا مقدار شش تایی رنگ مشخص می کند.
hspace	این ویژگی فضای افقی اطراف <code>marquee</code> را تعیین می کند. می تواند مقداری مانند 10 یا 20 درصد داشته باشد.
vspace	این ویژگی فضای عمودی اطراف <code>marquee</code> را تعیین می کند. می تواند مقداری مانند 10 یا 20 درصد داشته باشد.

در ادامه چند **مثال** از استفاده‌ی برچسب `marquee` می بینید.



```
<!DOCTYPE html>
<html>
<head>
  <title>HTML marquee Tag</title>
</head>
<body>
  <marquee>This is basic example of marquee</marquee>
</body>
</html>
```

مثال 2



```
<!DOCTYPE html>
<html>
<head>
  <title>HTML marquee Tag</title>
</head>
<body>
  <marquee width="50%">This example will take only 50% width</marquee>
</body>
</html>
```

مثال 3

```

<!DOCTYPE html>
<html>
<head>
    <title>HTML marquee Tag</title>
</head>
<body>
    <marquee direction="right">This text will scroll from left to right</marquee>
</body>
</html>

```

مثال 4

```

<!DOCTYPE html>
<html>
<head>
    <title>HTML marquee Tag</title>
</head>
<body>
    <marquee direction="up">This text will scroll from bottom to up</marquee>
</body>
</html>

```

سربرگ HTML

یاد گرفتیم که یک نمونه داکیومنت HTML دارای ساختار زیر می باشد.

Document declaration tag

```

<html>
<head>
    Document header related tags
</head>
<body>
    Document body related tags
</body>
</html>

```

این فصل جزئیات بیشتری در مورد بخش سربرگ در HTML ارائه می دهد که به وسیله ی برچسب **<head>** نمایش داده می شود. برچسب **<head>** حاوی برچسب های مهمی می باشد که عبارتند از **<title>**, **<meta>**, **<noscript>** tags. و **<link>**, **<base>**, **<style>**, **<script>**,

برچسب **<title>** در HTML

این برچسب برای تعیین تیتر داکیومنت HTML استفاده می شود. در زیر مثالی می بینید از ارائه ی تیتر به داکیومنت HTML.

```

<!DOCTYPE html>
<html>
<head>

```

112

```
<title>HTML Title Tag Example</title>
<head>
<body>
  <p>Hello, World!</p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید خواهد کرد.

Hello, World!

D A D E H A

برچسب <meta> در HTML

این برچسب برای ارائه‌ی متأذیتا در مورد داکیومنت HTML استفاده می‌شود که اطلاعاتی از قبیل انقضا صفحه، گردآورنده‌ی صفحه، لیست کلمات کلیدی، توصیف صفحه و غیره ارائه می‌دهد.

در ادامه استفاده‌های مهم برچسب <meta> در داکیومنت HTML ارائه شده‌اند.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Meta Tag Example</title>
  <!-- Provide list of keywords -->
  <meta name="keywords" content="C, C++, Java, PHP, Perl, Python">
  <!-- Provide description of the page -->
  <meta name="description" content="Simply Easy Learning by Tutorials Point">
  <!-- Author information -->
  <meta name="author" content="Tutorials Point">
  <!-- Page content type -->
  <meta http-equiv="content-type" content="text/html; charset=UTF-8">
  <!-- Page refreshing delay -->
  <meta http-equiv="refresh" content="30">
  <!-- Page expiry -->
  <meta http-equiv="expires" content="Wed, 21 June 2006 142527 GMT">
  <!-- Tag to tell robots not to index the content of a page -->
  <meta name="robots" content="noindex,nofollow">
</head>
<body>
  <p>Hello, World!</p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید می‌کند.

Hello, World!

برچسب <base> در HTML

این برچسب برای تعیین URL پایه برای همه URL های وابسته در صفحه استفاده می شود، که به این معناست که همه URL های دیگر هنگامی که برای آیتم ارائه شده قرار می گیرند، دیگر URL ها به پایه زنجیر خواهند شد.

به عنوان مثال تمام صفحات و تصاویر ارائه شده، پس از پیشوند دار کردن URL های ارائه شده با URL پایه مسیر prefixing جستجو خواهند شد.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Base Tag Example</title>
  <base href="http://www.tahlidadeh.com/" />
</head>
<body>
  
  <a href="/html/index.htm" title="HTML Tutorial" />HTML Tutorial</a>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید خواهد کرد.



برچسب <link> در HTML

این برچسب ارتباط بین داکیومنت جاری و منبع خارجی را تعیین می کند. در ادامه مثالی را می بینید از لینک یک فایل style sheet خارجی موجود در مسیر CSS با یک web root.

```
<!DOCTYPE html>
```

```

<html>
<head>
  <title>HTML link Tag Example</title>
  <base href="http://www.tahlildadeh.com/" />
  <link rel="stylesheet" type="text/css" href="/css/style.css">
</head>
<body>
  <p>Hello, World!</p>
</body>
</html>

```

این مثال نتیجه‌ی زیر را تولید خواهد کرد.

Hello, World!

برچسب `<style>` در `<html>`

این برچسب برای تعیین `style sheet` برای داکیومنت جاری `HTML` استفاده می‌شود. در ادامه **مثالی** را می‌بینید از تعریف برخی قوانین `style sheet` در داخل برچسب `<style>`.

```

<!DOCTYPE html>
<html>
<head>
  <title>HTML style Tag Example</title>
  <base href="http://www.tahlildadeh.com/" />
  <style type="text/css">
    .myclass {
      background-color: #aaa;
      padding :10px;
    }
  </style>
</head>
<body>
  <p class="myclass">Hello, World!</p>
</body>
</html>

```

این مثال نتیجه‌ی زیر را تولید خواهد کرد.

Hello, World!

توجه برای فراگیری چگونگی کار **Cascading Style Sheet** یک آموزش مجازی موجود در [اینجا](#) را کلیک کنید.

برچسب <script> در HTML

این برچسب برای وارد کردن فایل اسکریپت خارجی و یا تعریف فایل اسکریپت داخلی برای داکیومنت HTML استفاده می‌شود. در زیر مثالی را می‌بینید که در آن از جاوا اسکریپت برای تعریف عملکرد ساده‌ی جاوا اسکریپت استفاده می‌کنیم.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML script Tag Example</title>
<base href="http://www.tahlildadeh.com/" />
<script type="text/javascript">
function Hello() {
    alert("Hello, World");
}
</script>
</head>
<body>
<input type="button" onclick="Hello();" name="ok" value="OK" />
</body>
</html>
```

آموزش Cascading style sheet

بیان می‌کند که داکیومنت‌ها روی صفحه و در چاپ چگونه نشان داده می‌شوند، یا شاید چگونه بیان می‌شوند. از زمانی که کنسرسیووم در سال 1994 تاسیس شد، W3C به طور فعالانه ای در استفاده از Style sheet روی وب پیشرفت کرده است.

CSS جایگزین‌های ساده و موثری را برای تعیین ویژگی‌های مختلف برچسب‌های HTML ارائه می‌دهد. با استفاده از CSS شما می‌توانید تعدادی از ویژگی‌های طراحی را برای یک HTML ارائه شده تعیین کنید. هر ویژگی دارای نام مقدار می‌باشد که به وسیله‌ی علامت () از هم جدا شده‌اند. اطلاعیه‌ی مربوط به هر ویژگی نیز توسط علامت نقطه ویرگول (;) از یکدیگر جدا شده‌اند.

مثال

ابتدا اجازه بدھید که مثالی از داکیومنت HTML را مورد بررسی قرار دهیم که برای تعیین رنگ و اندازه‌ی فونت از برچسب استفاده می‌کند.

```
<!DOCTYPE html>
```

116

```

<html>
<head>
  <title>HTML CSS</title>
</head>
<body>
  <p><font color="green" size="5">Hello, World!</font></p>
</body>
</html>

```

می توانیم با استفاده از **style sheet** مثال بالا را مانند زیر بازنویسی کنیم.

```

<!DOCTYPE html>
<html>
<head>
  <title>HTML CSS</title>
</head>
<body>
  <p style="color:green;font-size:24px;">Hello, World!</p>
</body>
</html>

```

این مثال نتیجه‌ی زیر را تولید خواهد کرد.

Hello, World!

شما می توانید به سه روش از **CSS** در داکیومنت **HTML** خود استفاده کنید.

Style sheet خارجی – قوانین **style sheet** را در یک فایل **css** مجزا تعریف می کند، و سپس آن فایل را با

استفاده از برچسب **<link>** در **HTML** وارد داکیومنت **HTML** شما می کند.

Style sheet داخلی – قوانین **style sheet** را در با استفاده از برچسب **<style>** در بخش سربرگ داکیومنت **HTML** تعریف می کند.

Inline style sheet – قوانین **style sheet** را مستقیماً و به همراه عناصر **HTML** با استفاده از ویژگی **style** تعریف می کند.

اجازه بدھید هر سه مورد را یکی یکی و با استفاده از مثال‌های مناسب بررسی کنیم.

Style sheet خارجی

اگر می خواهید از style sheet خود در صفحات مختلف استفاده کنید، توصیه می شود که یک متداول در یک فایل مجزا تعریف کنید. یک فایل cascading style sheet دارای ضمیمه هایی مانند css می باشد و با استفاده از برچسب <link> وارد فایل های HTML خواهد شد.

مثال

توجه داشته باشید که ما یک فایل به نام style.css به دارای قوانین زیر می بشد، تعریف می کنیم.

```
.red{  
  color :red;  
}  
.thick{  
  font-size:20px;  
}  
.green{  
  Color:green;  
}
```



در اینجا سه قانون CSS را تعریف کردیم که برای سه گروه متفاوت تعریف شده در برچسب های HTML مناسب می باشند. پیشنهاد می کنم در مورد چگونگی تعریف این قوانین خود را اذیت نکنید، زیرا هنگام مطالعه آنها را فرا خواهید گرفت. اکنون اجازه بدھید از فایل CSS خارجی بالا در داکیومنت HTML زیر استفاده کنیم.

```
<!DOCTYPE html>  
<html>  
<head>  
  <title>HTML External CSS</title>  
  <link rel="stylesheet" type="text/css" href="/html/style.css">  
</head>  
<body>  
  <p class="red">This is red</p>  
  <p class="thick">This is thick</p>  
  <p class="green">This is green</p>  
  <p class="thick green">This is thick and green</p>  
</body>  
</html>
```

این مثال نتیجه ی زیر را تولید خواهد کرد.

This is red

This is thick

This is green

This is thick and green

Style sheet داخلی

اگر می خواهید قوانین **style sheet** را برای یک داکیومنت مجزا به کار ببرید، فقط آن موقع است که می توانید این قوانین را با استفاده از برچسب **<style>** وارد بخش سربرگ داکیومنت **HTML** کنید.

قوانین تعریف شده در **style sheet** داخلی، قوانین تعریف شده در فایل **CSS** خارجی را نیز در بر می گیرد.

مثال

اجازه بدھید **مثال** بالا را یک بار دیگر بازنویسی کنیم، اما در اینجا قوانین **style sheet** را در همان داکیومنت **HTML** و با استفاده از برچسب **<style>** می نویسیم.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Internal CSS</title>
  <style type="text/css">
    .red {
      color: red;
    }
    .thick {
      font-size :20px;
    }
    .green {
      color: green;
    }
  </style>
</head>
<body>
  <p class="red">This is red</p>
  <p class="thick">This is thick</p>
  <p class="green">This is green</p>
  <p class="thick green">This is thick and green</p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید خواهد کرد.

This is red

This is thick

This is green

This is thick and green

Inline Style Sheet

شما می توانید با استفاده از ویژگی **style** از برچسب مربوطه، قوانین **style sheet** را مستقیما برای هر عنصر **HTML** به کار ببرید. این امر فقط زمانی می تواند انجام شود که شما علاقمند به ایجاد تغییرات خاص در هر کدام از عناصر **HTML** می باشید.

قوانین تعریف شده با عنصر درون خطی، قوانین تعریف شده در یک فایل **CSS** خارجی و نیز قوانین تعریف شده در یک عنصر **<style>** را در برمی گیرد.

اجازه بدھید **مثال** بالا را یک بار دیگر بازنویسی کنیم، اما این بار قوانین **style sheet** را همراه با قوانین **HTML** و با استفاده از ویژگی **style** در عناصر خواهیم نوشت.

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Inline CSS</title>
</head>
<body>
  <p style="color:red;">This is red</p>
  <p style="font-size:20px;">This is thick</p>
  <p style="color:green;">This is green</p>
  <p style="color:green;font-size:20px;">This is thick and green</p>
</body>
</html>
```

این مثال نتیجه‌ی زیر را تولید خواهد کرد.

This is red

This is thick

This is green

This is thick and green

آموزش javascript

یک اسکریپت قطعه‌ی کوچکی از برنامه است که می‌تواند به وب سایت شما تعامل اضافه کند. به عنوان مثال یک اسکریپت می‌تواند یک هشدار پاپ آپ مربوط به باکس پیغام تولید کند، یا یک منوی رو به پایین ایجاد کند.

این اسکریپت توسط VBScript یا javascript نوشته می‌شود.

شما می‌توانید با استفاده از هر زبان اسکریپتی، کارکردهای متنوع کوچکی به نام گردانندگان رویداد بنویسید و سپس می‌توانید آن عملکردها را با استفاده از ویژگی‌های HTML اجرا کنید.

این روزها فقط javascript و چارچوب‌های منتظر با آن توسط بسیاری از توسعه دهندگان وب استفاده می‌شوند. VBScript حتی توسط برخی مرورگرها پشتیبانی هم نمی‌شود.

شما می‌توانید کد javascript را در یک فایل مجزا نگهداری کرده و سپس هر زمان که لازم بود آن را وارد کنید، یا می‌توانید قابلیت را در داخل خود داکیومنت HTML تعریف کنید. اجازه بدھید هر دو مورد را یکی یکی با مثال های مناسب بررسی کنیم.

جاوااسکریپت خارجی

اگر قصد دارید قابلیتی را تعریف کنید که در انواع داکیومنت‌های HTML استفاده می‌شود، بهتر است آن قابلیت را در یک فایل مجزای جاوااسکریپت حفظ کرده و آن فایل را وارد داکیومنت‌های HTML خود کنید. یک فایل جاوااسکریپت دارای ضمیمه‌هایی مانند `js` می‌باشد که با استفاده از برچسب `<script>` وارد فایل‌های HTML می‌شود.

مثال

تصویر کنید که با استفاده از جاوااسکریپت در `script.js` یک عملکرد کوچک را تعریف می‌کنیم که دارای کد زیر می‌باشد.

```
function Hello()
{
    alert("Hello, World");
}
```

اکنون اجازه بدھید از فایل جاواسکریپت خارجی بالا در داکیومنت HTML خود استفاده کنیم، مانند زیر

```
<!DOCTYPE html>
<html>
<head>
    <title>Javascript External Script</title>
    <script src="/html/script.js" type="text/javascript" /></script>
</head>
<body>
    <input type="button" onclick="Hello();" name="ok" value="Click Me" />
</body>
</html>
```

این مثال نتیجه ای را به دنبال خواهد داشت که می توانید با کلیک کردن بر روی دکمه **Click Me** ارائه شده توسط این مثال آن را امتحان کنید.

جاوا اسکریپت درونی

شما می توانید کد جاواسکریپت را مستقیماً روی داکیومنت HTML خود بنویسید. معمولاً کد اسکریپت را با استفاده از برچسب **<script>** در سربگ داکیومنت نگهداشی می کنیم در غیر این صورت هیچ محدودیتی وجود نداشته و می توانید کد خود را در هر جایی در داکیومنت به جز داخل برچسب **<script>** نگه دارید.

```
<!DOCTYPE html>
<html>
<head>
    <title>Javascript Internal Script</title>
    <script type="text/javascript">
        function Hello() {
            alert("Hello, World");
        }
    </script>
</head>
<body>
    <input type="button" onclick="Hello();" name="ok" value="Click Me" />
</body>
</html>
```

این **مثال** نتیجه ای را به دنبال خواهد داشت که می توانید با کلیک کردن بر روی دکمه می ارائه شده توسط این **مثال** آن را امتحان کنید.

گردانندگان رویدادها (EventHandler)

گردانندگان رویدادها عملکردهای تعریف شده ای ساده ای می باشند که می توانند بر علیه هر یک از عملکردهای ماوس یا صفحه کلید فراخوانده شوند. شما می توانید عملکرد خود را در یک گرداننده ای رویداد تعریف کنید که می تواند یک خط تا **1000** خط کد باشد.

در زیر مثالی را می بینید که چگونگی نگارش یک گرداننده ای رویداد را نشان می دهد. اجازه بدھید یک تیتر ساده ای (**EventHandler**) را در تیتر داکیومنت بنویسیم. هر زمان که کاربر ماوس را روی یک پاراگراف بیاورد، این عملکر را فرا خواهیم خواند.

```
<!DOCTYPE html>
<html>
<head>
<title>Event Handlers Example</title>
<script type="text/javascript">
function EventHandler() {
    alert("I'm event handler!!!");
}
</script>
</head>
<body>
<p onmouseover="EventHandler();">Bring your mouse here to see an alert</p>
</body>
</html>
```

این مثال نتیجه ای زیر را تولید خواهد کرد، ماوس خود را روی آن آورده تا نتیجه را ببینید.

Bring your mouse here to see an alert

مخفي کردن اسکریپت ها از مرورگرهای قدیمی

گرچه بیشتر مرورگرها این روزها جاوا اسکریپت را پشتیبانی می کنند، اما هنوز مرورگرهایی قدیمی تری هستند که این کار را نمی کنند. اگر مرورگری جاوا اسکریپت را پشتیبانی نکند، به جای اجرای اسکریپت کدی را به کاربر نمایش می دهد. برای جلوگیری از این مسئله می توانید کامنت های **HTML** را در اطراف جاوا اسکریپت قرار دهید، همانطور که در زیر می بینید.

JavaScript Example

```
<script type="text/javascript">
<!--
document.write("Hello Javascript!");
//-->
<script>
VBScript Example
<script type="text/vbscript">
<!--
document.write("Hello VBScript!")
'-->
<script>
```

عنصر `<noscript>`

می توانید یک سری اطلاعات جایگزین به کاربری ارائه دهید که مرورگر وی اسکریپت ها را پشتیبانی نمی کند، همچنین برای کاربرهایی که گزینه `script` مرورگر خود را غیرفعال کرده اند. شما می توانید اینکار را با استفاده از برجسب `<noscript>` انجام دهید.

JavaScript Example

```
<script type="text/javascript">
<!--
document.write("Hello Javascript!");
//-->
<script>
<noscript>Your browser does not support Javascript!
<noscript>
```

آموزشگاه تحلیلکر داده

دیالان ضمن مشکل از انتخاب شما، امیدواریم مطالب این کتاب برای شما منفید بوده باشد.

علاوه بر این می توانید پیشنهادات و انتخادات خود را از طریق ریلایانس Book.tahlildadeh@gmail.com با اداره میان گذارد.