

۷ کارنیل، بزرگترین شبکه موفقیت ایرانیان می باشد، که افرادی زیادی توانسته اند با آن به موفقیت برسند، فاطمه رتبه ۱۱ کنکور کارشناسی، محمد حسین رتبه ۶۸ کنکور کارشناسی، سپیده رتبه ۳ کنکور ارشد، مریم و همسرش راه اندازی تولیدی مانتو، امیر راه اندازی فروشگاه اینترنتی، کیوان پیوستن به تیم تراکتور سازی تبریز، میلاد پیوستن به تیم صبا، مهسا تحصیل در ایتالیا، و.... این موارد گوشه از افرادی بودند که با کارنیل به موفقیت رسیده اند، شما هم می توانید موفقیت خود را با کارنیل شروع کنید.

برای پیوستن به تیم کارنیلی های موفق روی لینک زیر کلیک کنید.

www.karnil.com

همچنین برای ورود به کانال تلگرام کارنیل روی لینک زیر کلیک کنید.

<https://telegram.me/karnil>

مقدمه ای بر HTML

در این درس با ویژگیهای فایل‌های HTML یا اچ‌تم‌ل و نحوه ایجاد یک فایل ساده و نمایش آن در صفحه مرورگرتان آشنا خواهید شد. (یادآوری: مرورگر، پروزر و Browser همگی یکی هستند و به نرم‌افزاری اطلاق میشود که به نمایش صفحات HTML پرداخته و واسطه کاربر با دنیای اینترنت است.)

یک فایل HTML چیست؟

- HTML را "اچ تی ام ال" و یا اچ‌تم‌ل بخوانید.
- HTML برگرفته از حروف اول **Language Markup Text Hyper**

میشود.

- یک فایل HTML فایلی از نوع text میباشد که متشکل از **markup tag** ها میباشد.
- مرورگر یا Browser از روی **markup tag** ها می‌فهمد که چگونه باید صفحه را نمایش بدهد.
- یک فایل HTML باید دارای انشعاب **htm** و یا **html** باشد.
- یک فایل HTML فایلی از نوع متن و text میباشد که با هر ویرایشگر و ادیتوری ساده ای قابل ایجاد است.
- وظیفه اصلی تگ‌های اچ‌تم‌ل (markup tags) بیان چگونگی نمایش اطلاعات میباشد.

میخواهید که یک فایل اچ‌تم‌ل بسازید؟

اگر از ویندوز مایکروسافت استفاده میکنید، Notepad را باز کرده (Mac کارها از SimplaeText استفاده کنند) و متن زیر را در آن تایپ کنید:

```
<html>

<head>
<title>Title of page</title>
</head>

<body>
This is my first html page. <b>This text is bold</b>
</body>

</html>
```

سپس فایل را با نام مثلا test.html در درایو C:\ ذخیره کنید.

برای دیدن فایل اچتمل بالا در مرورگر یا Brower تان (معمولاً اینترنت اکسپلورر و یا Firefox) کافی است که از روی منوی File/Open file فایل بالا را از روی محل ذخیره شده بخوانید. (به آدرس فایل در قسمت Address توجه کنید، مثلاً c:\test.html) یکی از مهمترین وظایف مرورگرها نمایش صفحات اچتمل میباشد، چه این صفحات روی کامپیوتر شما ذخیره شده باشند و یا اینکه روی سروری دیگر از اینترنت خوانده شوند. با کلیک روی این لینک فایلی شبیه فایل ذخیره شده توسط شما از روی وب سرور توسط مرورگرتان خوانده شده و سپس نمایش داده خواهد شد!!! (به آدرس فایل در قسمت Address توجه کنید، مثلاً <http://www.khaterat.com/html/dars/test.html>)

توضیح مثال بالا

اولین تگ مثال بالا تگ `<html>` میباشد. از روی این تگ، مرورگر نوع متن یعنی اچتمل بودنش را یافته و از روی `</html>` انتهای متن اچتمل را مییابد.

متن بین تگ `<head>` و تگ `</head>` اطلاعات شناسنامه ای یا "Header information" متن اچتمل بوده و نمایش داده نخواهند شد. در این مثال با کمک تگهای `<title>` و `</title>` تیترو یا Title صفحه که در این مثال عبارت "Title of page" میباشد در بالای مرورگر نمایش داده خواهد شد.

متن بین تگ `<body>` و انتهای آن یعنی تگ `</body>` تنها اطلاعاتی است که توسط مرورگر نمایش داده خواهند شد.

متن بین تگ `` و انتهای آن یعنی تگ `` بصورت Bold و یا توپر نمایش داده خواهند شد.

نام فایل test.html بوده ، در شاخه dars قرار گرفته و روی وب سروری که میزبان سایت khaterat.com است قرار دارد.

انشعاب فایل htm یا html ؟

اغلب مرورگرها هر دو نوع انشعاب را به خوبی میشناسند ولی ترجیحاً به هنگام ذخیره فایل‌های اچتمل از html استفاده کنید. (استفاده از انشعابهای سه حرفی مانند htm مربوط به قدیم و سیستم عامل‌هایی چون DOS بودند).

نکته ای در مورد ادیتورهای اچتمل (HTML Editors)

با وجودیکه با استفاده از ادیتورهای اختصاصی اچتمل مانند FrontPage و یا Claris Home Page امکان نوشتن و طراحی صفحات اچتمل بصورت WYSIWYG یا "Get You What Is See You What" وجود دارد اما توصیه میشود که از ادیتورهای معمولی متن برای تهیه صفحات خود استفاده کرده، عملکرد تگها را شناخته و هیچگاه بدون اینکه بدانید کاربرد یک تگ چیست آنرا بکار نبرید. (ادیتورها

همان برداشگرهای متن هستند.)

بیشترین سوالات پرسیده شده (FAQ) :

فایلم را درست کردم ولی هنوز نمیتوانم آنرا در صفحه مرورگرم ببینم، مشکل کجاست؟
مطمئن شوید که فایل را با انشعاب درست (html یا htm) ذخیره کرده اید. در ضمن مطمئن شوید که همان فایل را Open کرده اید. نام و مسیر فایل را در قسمت Address مرورگرتان چک کنید.

هر بار پس از تغییر و ادیت مثالها نسخه اولیه را در صفحه مرورگرم میبینم و تغییرات داده شده مشاهده نمیشود، مشکل کجاست؟

به خاطر بالا بردن سرعت، همیشه مرورگرها از نسخه های موجود در Cache یا حافظه موقت خود برای خواندن صفحات استفاده میکنند. برای وادار کردن مرورگر به خواندن اصل صفحه کافی است که در مرورگرتان Refresh/Reload کنید. در اینترنت اکسپلورر کلید F5 یا View/Refresh و در فایرفاکس Ctrl+R یا View/Reload اینکار را انجام خواهند داد.

آیا میتوانم از هر دو مرورگر Internet Explorer و Firefox استفاده کنم؟

بله، فقط اگر از استاندارد یونیکد برای فارسی نویسی استفاده میکنید، برای دیدن درست صفحات فارسی باید از نسخه ۵ Internet Explorer و به بالا و در مورد Firefox از نسخه ۱.۰ و به بالا استفاده کنید.

کجا میتوانم آخرین نسخه از مرورگرهای Internet Explorer و Firefox را دانلود کنم؟
Internet Explorer اینجاست و Firefox اینجا.

فایلهای HTML ام را چگونه به روی اینترنت منتقل کنم؟

باید از نرم افزارهای FTP که وظیفه انتقال فایلها در اینترنت را دارند کمک بگیرید.

WWW & HTML

در این درس با مفاهیم وب، اینترنت، مرورگرها و استانداردهای وب آشنا خواهید شد.

منظور از وب چیست؟

- وب شبکه ای است متشکل از تمامی کامپیوترهای دنیا، شبکه ای از شبکه ها.
- اینترنت، وب، WWW، web یا Web Wide World همگی یک چیزند.
- تمامی کامپیوترهای وب میتوانند با هم ارتباط داشته باشند.
- کامپیوترهای موجود در وب با کمک استاندارد ارتباطی یا پروتوکل HTTP با هم ارتباط برقرار میکنند .

Learn HTML and CSS with w3schools | 2012

نحوه کارکرد وب چگونه است؟

- اطلاعات صفحات وب داخل فایل‌هایی به نام Web Pages و یا متن‌های وب (HTML documents) قرار دارند.
- این فایل‌ها یا صفحات روی Web Server یا کامپیوترهای سرورس دهنده وب ذخیره شده‌اند.
- برای دیدن صفحات وب از نرم‌افزاری به نام مرورگر و یا Web Browser استفاده میشود.
- سرورهای وب یا Web Servers کامپیوترهای هستند که با کمک استاندارد ارتباطی HTTP به تبادل اطلاعات با مرورگرها (web Clients) می‌پردازند.
- مرورگرهای وب را Web browsers یا web Clients هم مینامند.
- دو مرورگر Internet Explorer و Firefox جزو معروفترین مرورگرها حساب میشوند.
- مرورگر Internet Explorer متعلق به شرکت مایکروسافت و مرورگر Firefox متعلق به شرکت Mozilla میباشد.
- مرورگر Firefox دارای امنیت بیشتری بوه و بصورت open Source میباشد و نسبت به Internet Explorer برتر محسوب میشود.
- مرورگر Internet Explorer گاهی مختصراً IE نامیده میشود.
- اسامی بعضی از مرورگرهای دیگر به قرار روبرو است: Mosaic, Netscape Navigator, Opera, Mozilla, Safari(Mac computers)
- امیدوارم که شما طراح و برنامه‌نویس اولین مرورگر تماماً فارسی باشید!

مرورگرها چگونه به خواندن صفحات وب می‌پردازند؟

- یک مرورگر با کمک یک Request درخواستی برای خواندن یک صفحه از وب سرور میکند.
- این Request یا درخواست بر اساس استاندارد ارتباطی یا پروتوکل HTTP بوده و شامل آدرس صفحه مورد نظر میباشد.
- آدرس یک صفحه وب چیزی شبیه `http://www.khaterat.com/www/htmlwww.php` است.
- بخش `http://` نوع پروتوکل و یا استاندارد ارتباطی را تعیین میکند، `www.khaterat.com` نام دومین یا Domain است و `/www/htmlwww.php` مسیر و نام صفحه‌ای است که باید خوانده شود.

مرورگرها چگونه صفحات وب را نمایش میدهند؟

- چگونگی نمایش یک صفحه وب بصورت مستتر در آن وجود دارد.(اگر شما منظورم را فهمیدید من را هم خبردار کنید!)
- مرورگرها از روی دستورالعمل‌های داخل صفحات وب و با کمک تگ‌ها به نمایش صفحات می‌پردازند.
- وظیفه اصلی تگ‌های اجتملی (HTML tags) بیان نحوه چین و قرارگرفتن اطلاعات است و ظاهر نمایش با CSS میباشد.
- یک تگ اجتملی چیزی شبیه `<p> abc </p>` است.

چه کسانی استانداردهای وب را تعیین میکنند؟

- تعیین استانداردهای وب ربطی به شرکتهای مایکروسافت و یا Mozilla ندارد.
- World Wide Web Consortium یا W3C متولی بین المللی تعیین استانداردهای وب است.
- HTML, CSS and XML از مهمترین استانداردهای تصویب شده وب میباشند.
- برای دیدن فهرست تمامی استانداردهای تعریف شده توسط W3C به این صفحه از دایرالمعارف WIKI مراجعه کنید: W3C

عناصر (Elements)

یک فایل اچتمل متنی از نوع text است که از عناصر یا element ها تشکیل شده و برای ایجاد عناصر از تگ ها یا tags استفاده میشود. در این فصل با عناصر و تگ ها اچتمل آشنا خواهید شد.

تگ های اچتمل (HTML Tags)

- با کمک تگ های اچتمل عناصر یا Elements ساخته میشوند.
- در زبان اچتمل حدود ۸۰ عنصر تعریف شده است.
- تگ های اچتمل بوسیله دو نویسه (char) < و > ساخته میشوند.
- تگ های اچتمل معمولاً بصورت زوج ظاهر میشوند، مانند test
- تگ اول در یک زوج تگ مثلاً تگ شروع و تگ دوم مثلاً تگ پایانی نام دارد.
- متن بین تگ اول و تگ دوم در یک زوج تگ محتوای عنصر یا element نامیده میشود، مثلاً "test"
- تگ های اچتمل را میتوانید بوسیله حروف لاتین کوچک (lower case) و یا بزرگ (upper case) بنویسید و case sensitive نیستند. برای مثال دو تگ و معادل هم هستند ولی شدیداً توصیه میشود که به خاطر سازگاری با XHTML از حروف کوچک استفاده شود.

عناصر اچتمل (HTML Elements)

مثال بخش مقدمه را در نظر بگیرید:

```
<html>

<head>
<title>Title of page</title>
</head>

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

```
</html>
```

*** نمونه ای از یک عنصر اچتمل:

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

عنصر اچتمل بالا با تگ شروع شده و با تگ پایان می یابد. محتوای این عنصر عبارت "This text is bold" است. لازم به ذکر است که کاربرد تگ نمایش توپر یا bold متون است.

*** مثال دوم از یک عنصر اچتمل :

```
<body>
```

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

```
</body>
```

عنصر اچتمل بالا با تگ <body> شروع شده و با تگ </body> پایان می یابد. همانطور که میبینید گاهی یک عنصر حاوی یک یا چند تگ دیگر میباشد. وظیفه تگ <body> تعیین بدنه اصلی یا body یک متن اچتمل است. لازم به یادآوری است که تنها اطلاعات بخش <body> یک فایل اچتمل در صفحه مرورگر نمایش داده خواهد شد.

شناسه های یک تگ (Tag Attributes)

تگ ها میتوانند حاوی اطلاعات اضافی دیگر باشند، به این اطلاعات شناسه یا Attribute میگویند و وظیفه آنها بیان دیگر اطلاعات یک عنصر یا Element میباشد. مثلا در مورد تگ <body> شناسه ای به نام bgcolor وجود دارد که رنگ زمینه متن (background) را تعیین میکند برای نمونه اگر میخواهید که رنگ زمینه صفحه اچتملتان سیاه باشد کافی است که به شکل زیر عمل کنید :

```
<body bgcolor="black">
```

در مثال زیر تگ <table> دارای چهار شناسه مختلف با نامهای width, height, align, border و مقادیر 100, 60, center و 1 میباشد.

```
<table border="0" width="100" height="60" align="center" >
```

```
...
```

```
</table>
```

شناسه ها به صورت کلی "مقدار=نام" یا "name=value" نوشته میشوند و همیشه به تگ شروع یک عنصر یا Element اضافه میشوند و نهایتا اگر در یک عنصر یا Element شناسه ها قید نشوند از مقادیر قراردادی یا default آنها استفاده خواهد شد مثلا در تگ body اگر شناسه bgcolor نوشته نشود از رنگ سفید برای زمینه صفحه استفاده خواهد شد. مقدار یک شناسه را میتوانید داخل نویسه های " و یا ' بنویسید و اختیار دست شما است فقط در مواردی که مقدار یک شناسه شامل نویسه " هم میشود باید از نویسه ' استفاده شود.

تگ های اصلی (Basic Tags)

تگ ها (Tags) اجزا تشکیل دهنده عناصر یا elements های اچتمل هستند و در این فصل با عناصر و تگ های پایه و مهم:

<p> , <!-- comment --!> , <h6> <h1> ,
 , <hr>

آشنا خواهید شد. یکی از بهترین روش های یادگیری تگ های اچتمل دیدن مثالها و تغییر آنها میباشد و با کمک ادیتور اختصاصی ما و با استفاده از امکانات فارسی نویسی آن به صورت آنلاین میتوانید به مطالعه و یادگیری مثالها پرداخته، کدهای اچتمل را تغییر داده و با کلیک روی دکمه "نمایش نتایج" به مشاهده نتایج بپردازید.

مثالها :

مثالی ساده از یک صفحه اچتمل در این مثال بسیار ساده عبارت معروف "Hello World!" در صفحه مرور گر نمایش داده شده و در واقع محتوای عنصر body تماما نمایش داده خواهد شد.

پاراگرافهای ساده:

چگونگی نمایش متون با کمک تگ <p> و بصورت پاراگرافی

نمایش متون فارسی:

چگونگی نمایش متون فارسی. برای جزئیات بیشتر به فصل فارسی نویسی مراجعه کنید.

مثالهای بیشتر:

سر تیترها (Headings)

سر تیترها با کمک تگ های <h1> تا <h6> تعریف میشوند. <h1> معرف بزرگترین سر تیتر و <h6> معرف کوچکترین سر تیتر است. مرورگر به هنگام نمایش یک سر تیتر بصورت اتوماتیک یک سطر خالی قبل و بعد از هر سر تیتر اضافه خواهد کرد.

```
<h1>This is a heading</h1>
<h2>This is a heading</h2>
<h3>This is a heading</h3>
<h4>This is a heading</h4>
<h5>This is a heading</h5>
<h6>This is a heading</h6>
```

مثال بالا در صفحه مرورگر بدین شکل نمایش داده خواهد شد.

پاراگرافها (Paragraphs)

پاراگرافها با کمک تگ <p> معرفی میشوند. مرورگرها به هنگام نمایش یک پاراگراف بصورت اتوماتیک یک سطر خالی قبل و بعد از آن نمایش خواهند داد.

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

مثال بالا در صفحه مرورگر بدین شکل نمایش داده خواهد شد.

سطر جدید (Line Breaks)

برای رفتن سر سطر جدید از تگ `
` استفاده میشود. در اینحالت یک پاراگراف جدید ایجاد نمیشود. تگ `
` از نوع تگ های خالی بوده و دارای تگ انتهائی (مثلا `</br>`) نمیباشد.

```
<p>This <br> is a para<br>graph with line breaks</p>
```

مثال بالا در صفحه مرورگر بدین شکل نمایش داده خواهد شد.

کامنت ها در اچتمل (Comments)

برای نوشتن شرح و توضیحات در مورد کدهای اچتمل باید از تگ خاصی استفاده کنید. برای اینکار باید متن و شرحتان را درون علائم `<!-->` و `!</-->` قرار دهید. مرورگر تگ های comment را در نظر نگرفته و محتوی آنها را نمایش نخواهد داد و فقط شرح و توضیحات برای برنامه نویس و دیگر افرادی که احتمالاً در آینده با کد اچتمل کار خواهند کرد مفید خواهد بود. (به محل نویسه "!" توجه کنید! فقط یکی و آنهم در ابتدا)

```
<!-- This is a comment -->
```

مثالی در مورد comments و نحوه استفاده از آن

چند نکته کاربردی:

توجه داشته باشید که بدلیل وجود مرورگرهای متفاوت (اکسپلورر، نت اسکپ، فایرفاکس، صفری،...) و به دلیل تفاوت دقت نمایش صفحه نمایش کامپیوترها، صفحات اچتمل با اندکی تفاوت در حالتهای مختلف نمایش داده میشوند و همیشه سعی کنید که صفحاتتان را نه تنها با اکسپلورر که معروفترین مرورگر است بلکه با فایرفاکس و نت اسکپ و حتی مرورگرهای کامپیوترهای مکینتاش چک کرده و همچنین در دقت های نمایش `x6000۸۰۰` و `X768۱۰۲۴` آن را امتحان کنید.

از نظر فاصله و سطر بندی، متنی که در صفحه ادیتورتان تایپ میکنید با چیزی که مرورگر نمایش خواهد داد متفاوت خواهد بود. همیشه به یاد داشته باشید که فاصله های اضافی (space) و خطهای خالی متن در صفحه ادیتور توسط مرورگر در نظر گرفته نشده و نمایش داده نخواهد شد.

برای نمایش بیش از یک فاصله خالی باید از نویسه ها یا ترکیب کاراکتری خاصی (None Breaking Space) استفاده کنید. در بخش (Entities) نهادها به این مطلب بیشتر پرداخته خواهد شد.

فاصله های اضافی بین کلمات در یک متن اچتمل توسط مرورگرها دیده نخواهد شد و در نمایش همیشه تبدیل به تنها یک فاصله (space) خواهد شد. در ضمن یک خط خالی در متن ادیتور بصورت یک فاصله یا space نمایش داده خواهد شد.

برای ایجاد یک سطر جدید هیچگاه از یک تگ `<p>` خالی استفاده نکنید و به جای آن از تگ `
` استفاده کنید..

مرورگرها به هنگام نمایش بعضی عناصر بصورت اتوماتیک یک سطر خالی قبل و بعد از آن عنصر نمایش خواهند داد. برای نمونه این گروه از عنصرها میتوان از پاراگراف (<p>) و سرتیترها (<h>...) نام برد، با کمک استایل شیتها میتوانید این پیش فرضها را تغییر دهید.

تگ <hr> یا **Roler Horizontal** سبب نمایش یک خط افقی خواهد شد و در واقع بخش های مختلف مطالب این صفحات با کمک این تگ از هم جدا شده اند.

مثالهای بیشتر:

باز هم مثال پاراگرافها

این مثال بعضی از مقادیر قراردادی عناصر پاراگراف را شرح میدهد.

ترازبندی پاراگرافها

این مثال نحوه تراز بندی یک پاراگراف را با کمک شناسه یا attribute ای به نام align نشان میدهد. در این حالت پاراگراف در وسط ناحیه نمایش داده خواهد شد.

(align="center")

رفتن سر سطر جدید

این مثال نحوه استفاده از تگ
 را در یک متن اچتمل نشان میدهد.

سر تیترها

این مثال نحوه استفاده از تگهای <h1> <h6> را در یک متن اچتمل نشان میدهد.

ترازبندی سر تیترها

این مثال نحوه تراز بندی یک سر تیتر را با کمک شناسه یا attribute ای به نام align نشان میدهد. در این حالت سر تیتر در وسط ناحیه نمایش داده خواهد شد.

(align="center")

نمایش خطوط افقی

این مثال نحوه نمایش خطوط افقی با استفاده از تگ
 میباشد.

استفاده از comment

این مثال نحوه استفاده از comment در کدهای اچتمل را نمایش میدهد.

تعیین رنگ زمینه صفحه

این مثال نحوه تعیین رنگ زمینه صفحه را با کمک یکی از شناسه های (attribute) عنصر body به نام bgcolor نشان میدهد.

(Background Color)

(bgcolor ="blue")

تگهای اصلی

در جدول زیر عناصر معرفی شده در این فصل به همراه لینکهای مربوطه جهت مطالعه بیشتر آورده شده است. توجه داشته باشید که برای هر عنصر فهرستی از شناسه ها یا Attributes موجود است و همچنین به شناسه های کنار گذاشته شده (Deprecated) در نسخه های آینده اجتملی توجه داشته باشید و سعی کنید که از آنها استفاده نکنید.

:

Start Tag	Purpose	کاربرد
<html>	Defines a html document	نشان شروع متن اجتملی
<body>	Defines the document's body	تعیین بدنه و قسمت اصلی صفحه اجتملی
<h1>-<h6>	Defines heading 1 to heading 6	تعریف سر تیترهای h1 تا h6
<p>	Defines a paragraph	تعریف پاراگراف
 	Inserts a single line break	رفتن سر خط جدید
<hr>	Defines a horizontal rule	نمایش خط افقی
<!-->	Defines a comment in the HTML source code	نوشتن شرح و comment

مرجع : HTML 4.01 Specification - فهرست عناصر - فهرست شناسه ها

فارسی نویسی

در این درس با فارسی نویسی، فارسی سازی و استاندارد یونیکد و نحوه استفاده از ادیتور آنلاین و فارسی این سایت آشنا خواهید شد. همچنین متاتگ http-equiv و شناسه dir نیز معرفی خواهد شد.

درس قبلی، درس بعدی

سؤالات مربوط به فارسی نویسی را میتوان به سه گروه عمده تقسیم کرد:

- چگونه فارسی بنویسم؟
- چگونه فارسی بخوانم؟
- چگونه صفحات اجتملی فارسی را نمایش دهم؟

در مورد فارسی نویسی و فارسی خوانی اگر از ادیتور آنلاین ما در بخش مثالها استفاده میکنید نیازی به فارسی سازی ندارید و با کمک ادیتور مزبور باید به راحتی قادر به تایپ فارسی و لاتین باشید. اگر میخواهید که در هر حال کامپیوترتان را فارسی کنید، مطالعه صفحات زیر راهنمای بسیار خوبی برای فارسی سازی میباشد:

- * راهنمای فارسی سازی ویندوز
- * قلم و صفحه کلید استاندارد
- * unicode.org
- * پروژه فارسی وب
- * وبلاگ فارسی FAQ

یونی کد چیست؟

به طور خلاصه و با صرف نظر از بعضی از پیچیدگیها می شود گفت یونی کد مجموعه ای بسیار بزرگ از نویسه (character) های مختلف است. این مجموعه تقریباً تمام نویسه های مورد استفاده در کامپیوتر در جهان را در خود دارد. از حرف سین الفبای فارسی گرفته تا حروف الفبای چینی و انواع و اقسام علامتها و حتی خط میخی فارسی باستان. مهمترین خاصیت یونی کد این است که به هر کدام از این نویسه ها یک شماره یکتا اختصاص می دهد که به آن کد یونی کد آن نویسه می گویند. کد یونی کد ی فارسی U+06CC است در حالی که کد یونی کد ی عربی U+064A است. وقتی یک پرونده ذخیره می شود این کدها هستند که ذخیره می شوند نه شکل نویسه ها. بنابر این تأثیر این مسأله این است که اگر پرونده ای با قالب یونی کد ذخیره شود و در هر کجای دیگر دنیا که از یونی کد استفاده می کند باز شود، همان نویسه ها درونش قرار خواهند داشت.

دو نکته بسیار مهم در مورد فارسی نویسی:

• روش کدگذاری UTF-8

اکیدا توصیه میشود که برای تایپ و نمایش متون فارسی از استاندارد یونیکد و روش کدگذاری UTF-8 استفاده شود و این روشی است که مطالب این سایت و ادیتور آنلاینش از آن استفاده میکنند. نکته مهم در نمایش صحیح متون فارسی تعیین نوع کدگذاری یا `encoding` صفحات فارسی است که برای اینکار باید از متاتگ خاصی به نام `http-equiv` در بخش `head` صفحات اچتمل به شکل زیر استفاده شود:

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

در اینصورت مرورگر یا Browser قبل از نمایش صفحه از روی متاتگ فوق نوع کدگذاری (Encoding) را تشخیص داده و دیگر مثلاً نیازی به تعیین دستی (دراکسپلورور ۵ : UTF-8 / View/Encoing/Unicode) نوع `Encoing` توسط بازدیدکننده سایت نخواهد بود.

• شناسه یا attribute ی به نام dir

سمت و جهت نمایش متون لاتین به صورت پیش فرض و `default` از چپ به راست (`Right To Left` یا `ltr`) میباشد. در مورد متون و جملات فارسی باید با کمک روشی جهت نمایش پیش فرض را به "راست به چپ" تغییر دهید. شناسه یا attribute ی به نام `dir` یا همان `direction` این کار را برای شما انجام میدهد. این شناسه دارای دو مقدار ممکن میباشد :

چپ به راست - `dir="ltr"` , Left-to-right text.

راست به چپ - `dir="rtl"` , Right-to-left text.

مثلا برای نمایش جمله فارسی "سلام به دنیای وب!" با کمک عناصر `p` یا `div` کافی است که به یکی از شکل‌های زیر عمل شود:

```
<p dir="rtl"> سلام به دنیای وب </p>
<div dir="rtl"> سلام به دنیای وب </div>
```

در بسیاری از عناصر و تگ‌های اچ‌تم‌ل امکان استفاده از شناسه `dir` میسر بوده و به عنوان نمونه میتوان از عناصر زیر نام برد:

```
<p> , <div> , <html> , <body> , <table> , <tr> , <td> , <h1 ... h6> , <input> , <pre> , <select> ,
<span> , ...
```

مثال‌های زیر نحوه ترازبندی متون فارسی را به نمایش می‌گذارند و برای مطالعه بیشتر در مورد ترازبندی متون "راست به چپ" به این صفحه مراجعه کنید. (عناصر `Span`، `BDO` و شناسه `lang` هم برای مطالعه مفید میباشند).

مثال‌های فارسی نویسی:

مثال ۱ : سلام ...

به بخش `head` و تگ `div` توجه کنید. متا تگ `http-equiv` نوع کدگذاری را تعیین میکند و در اینجا با کمک تگ `div` سمت و جهت نوشتن مطالب (از راست به چپ یا بالعکس) تعیین میگردد. `dir` به معنای `direction` و جهت بوده و `rtl` مخفف `to left right` و `ltr` معادل `left to right` میباشد.

مثال ۲ : متن‌ها و پاراگراف بندی

به عناصر `div` و `p` توجه کنید. با کمک شناسه `dir` جهت نوشتن مطالب (از راست به چپ یا بالعکس) تعیین میگردد.

مثال ۳ : متون فارسی و لاتین در کنار هم

مثالی کاملتر با راهنمای استفاده از عنصر `div` و شناسه `dir` آن

مثال ۴ : شناسه `dir` در عنصر `html`

اگر شناسه `dir` را در عنصر `html` بکار ببرید جهت نمایش تمامی متون از راست به چپ شده و برای نوشتن از چپ به راست باید به صورت موردی شناسه `dir="ltr"` را استفاده کنید. همچنین در این حالت ماسماک! `Scroll down` مرورگر در سمت چپ صفحه نمایش داده خواهد شد. در صورت استفاده شناسه `dir` در عنصر `body` ماسماک همان سمت راست خواهد ماند.

مثال ۵ : مثال کامل

نحوه استفاده شناسه `dir` در عنصرهای مختلف

مثال ۶ : ... فقط تصور کن ...

استفاده از شناسه `dir` در جداول

فرمت دهی (Formatting)

گروهی از تگ های اچتمل به فرمت دهی و تعیین شکل نمایش متون (مثلا ضخیم یا bold بودن، مورب یا ایتالیک بودن) میپردازند. (به یاد داشته باشد که در نسخه ۴ اچتمل این خواص به استایل شیتها محول شده است.) در این درس با این گروه از تگهای formating آشنا خواهید شد:

 , <big> , <code> , , , <i> , <ins> , <kbd> , <pre> , <s> , <samp> , <small> , <strike> , , <sub> , <sup> , <tt> , <u> , <var>

مثالهای شکل دهی متون (Text Formatting)

در مثالهای زیر نحوه استفاده از عناصر و تگهای شکل دهی متون شرح داده شده اند:

شکل دهی متون

شکلهای مختلف نمایش متون با کمک عناصر **b** , **strong** , **big** , **em** , **i** , **small** , **sub** , **sup**

متون از قبل شکل دهی شده

نمایش متون از قبل شکل داده شده با کمک عنصر **pre** یا **pre Formatted** برای نمایش اشعار فارسی استفاده از عنصر **pre** بهترین انتخاب میباشد.

عناصر دیگر شکل دهی متن

نمایش متون **underline**، **deleted** و ... با کمک عناصر **code** , **kbd** , **tt** , **samp** , **var** , **del** , **ins**

نحوه دیدن سورس کدهای اچتمل (HTML Source)

- اگر میخواهید که سورس یا اصل کدهای اچتمل صفحات وب را ببینید کافی است که در منوی View مرورگر تان حالت Source یا Page Source را انتخاب کرده و مرورگر کد اچتمل را درون ادیتوری باز کرده و قابل ذخیره توسط شما برای استفاده های بعدی میباشد.
- اگر صفحه مورد نظر از مجموعه فریمها (Frameset) استفاده کرده باشد برای دیدن سورس کد اصلی فریمها کافی است که ابتدا با کمک روش بالا به مشاهده سورس کدهای اچتمل پرداخته (لازم به توضیح است که در این حالت فقط تعاریف کلی فریمها قرار دارند.) و سپس برای دیدن کد اچتمل اختصاصی هر صفحه باید پس از "Right Click" روی هر صفحه "View Source" کرده و کد اچتمل هر فریم را جداگانه ببینید.
- اگر طراح سایت با کمک Java Script مانع دیدن سورس توسط شما میشود، ابتدا در منوی Tools/Internet Options.../Security/Custom Level مرورگرهای اکسپلورر امکانات جاواسکریپت مرورگر را غیر فعال کرده (disable) و سپس مراحل بالا را تکرار کنید!!!

چند نکته کاربردی:

- I. در استفاده از Underline دقت کنید چون در اکثر مرورگرها، پیوندها با این خطوط زیری نشان داده می شوند و این مساله میتواند کاربران را دچار اشتباه کند .
- II. به کارگیری تگهای EM و STRONG که جایگزین تگهای I و B هستند ، این اطمینان را می دهد که صفحه شما توسط همه مرورگرها بصورت صحیح نمایش داده خواهد شد .
- III. به ترتیب بسته شدن تگهای پایانی در مثال زیر دقت کنید، همان گونه که مشاهده می کنید عنصر STRONG بعد از عنصر U شروع شده است ، به همین خاطر تگ خاتمه دهنده آن قبل از تگ خاتمه دهنده U قرار می گیرد.

```
<U>version <STRONG>2.0</STRONG></U>
```

تگهای فرمت دهی متون

Start Tag	Purpose	کاربرد
	Defines bold text	نمایش توپر
<big>	Defines big text	نمایش در اندازه بزرگ
	Defines emphasized text	نمایش بصورت تاکید شده
<i>	Defines italic text	نمایش ایتالیک یا کج
<small>	Defines small text	نمایش در اندازه کوچک
	Defines strong text	نمایش قوی یا همان Bold
<sub>	Defines subscripted text	نمایش پایین تر از خط افقی
<sup>	Defines superscripted text	نمایش بالاتر از خط افقی
<ins>	Defines inserted text	نمایش به صورت خط زیر
	Defines deleted text	نمایش به صورت خط خورده
<s>	Deprecated. Use instead	از رده خارج
<strike>	Deprecated. Use instead	از رده خارج
<u>	Deprecated. Use styles instead	از رده خارج
Start Tag	Purpose	کاربرد
<code>	Defines computer code text	نمایش کدهای برنامه نویسی
<kbd>	Defines keyboard text	نمایش متن بصورت صفحه کلیدی
<samp>	Defines sample computer code	نمایش کدهای نمونه برنامه نویسی
<tt>	Defines teletype text	نمایش تله تایپ

Learn HTML and CSS with w3schools | 2012

<var>	Defines a variable	نمایش متغیرها
<pre>	Defines preformatted text	نمایش متون از قبل فرمت شده

مرجع : HTML 4.01 Specification - فهرست عناصر - فهرست شناسه ها

Character Entities

در استاندارد اچتمل گروهی از نویسه ها (Character) دارای معنی خاصی بوده (مثلا نویسه < شروع یک تگ و نویسه > انتهای یک تگ را نشان میدهد.) و برای نمایش آنها در صفحاتتان مستقیما قابل مصرف نمیباشند. در این فصل با فهرست این نویسه ها آشنا شده و نحوه درج آنها را با کمک Character Entities فرا خواهید گرفت.

درس قبلی، درس بعدی

Character Entities

در اچتمل بعضی از نویسه ها دارای معنی خاصی بوده و برای نمایش آنها در صفحاتتان باید از برابرهایی خاصی استفاده کنید. یک character entities از سه قسمت تشکیل میشود:

۱- نویسه ampersand (&)

۲- نام entity یا نویسه # و سپس شماره عددی entity

۳- و نهایتا نویسه semicolon (;)

برای مثال برای نمایش نویسه < دو روش وجود دارد، میتوانید از **<** یا از **<** استفاده کنید. در این مثال نام entity و 60 شماره عددی آن میباشد.

مزیت استفاده از نام entity در مقابل استفاده از شماره عددی آن سهولت یادآوری آن از روی نامش میباشد و عیب آن این است که تمامی مرورگرها نام گذاریهایی موجود برای Entity ها را قبول ندارند ولی در عوض همگی شماره های عددی entity ها را بخوبی میشناسند. توجه داشته باشید که entity ها به کوچک و بزرگی حروف حساس هستند و در واقع Case Sensitive میباشد.

پر کاربردترین character entity ها در اچتمل، نویسه فاصله یا Space یا Blank میباشد و نام رسمی آن non-breaking space میباشد. همانطور که میدانید مرورگرها فاصله های اضافی در متن را در نظر نگرفته (truncate spaces) و مثلا از ۱۰ نویسه فاصله، ۹ تای آنرا نادیده میگیرند و برای نمایش فاصله های اضافی باید از ** ** استفاده کنید.

فهرست پرکاربردترین Character Entities ها

Result (نمایش)	Description (شرح)	Entity Name (نام)	Entity Number (شماره عددی)
<	non-breaking space	 	
<	less than	<	<

>	greater than	>	>
&	ampersand	&	&
"	quotation mark	"	"
'	apostrophe		'

چند Character Entitie با کاربردهای کمتر

Result	Description	Entity Name	Entity Number
¢	cent	¢	¢
£	pound	£	£
¥	yen	¥	¥
§	section	§	§
©	copyright	©	©
®	registered trademark	®	®
×	multiplication	×	×
÷	division	÷	÷

برای دیدن فهرست کل Character entity ها در اچتمل ۴ سری به این صفحه بزنید.

پیوندهای اچتمل (HTML Links)

در این درس با پیوندها (Links) ، تگ های مربوطه مخصوصا تگ Anchor یا A و نحوه به کار گیری آنها آشنا خواهید شد، به یاد داشته باشید که با کمک پیوندها است که صفحات وب به هم مرتبط میشوند و همانند تاروی نامرئی اطلاعات و صفحات وب را به هم گره میزنند. همچنین شناسه های بسیار مهم href، target، و name تشریح خواهند شد.

درس قبلی، درس بعدی

در محیط وب، صفحات اچتمل با کمک پیوندها به یکدیگر متصل (Link) میشوند. اصطلاح ابرمتن (Hyper Text) در مقابل متن خطی (Linear) قرار دارد. در یک متن معمولی خواندن به شکل خطی و از ابتدا به انتها مییاشد و در مقابل در یک متن مختلط (Hyper) با کمک پیوندها میتوان از یک متن به هر صفحه دیگر در وب متصل شد. این کار با کمک عنصری معلوم الحال! به نام A یا Anchor میسر میگردد.

مثالها

ایجاد پیوندها با کمک متن

این مثال چگونگی ایجاد یک پیوند، لینک یا Link را در یک متن اچتمل نشان میدهد.

ایجاد پیوندها با کمک تصاویر

این مثال چگونگی ایجاد یک پیوند، لینک یا Link را با کمک یک تصویر (image) نشان میدهد.

مثالهای بیشتر

تگ Anchor و شناسه href

برای ایجاد پیوند به صفحات دیگر از تگ `<a>` استفاده میشود. پیوندها میتوانند به بخش دیگری از همان صفحه، صفحات دیگر وب، تصاویر، فایل‌های صوتی یا حتی فیلم‌ها و ... اشاره کنند. (در واقع به هر آدرس اینترنتی) فرم کلی یک پیوند به قرار زیر است:

```
<a href="url">Some Text</a>
```

در مثال بالا تگ `<a>` برای ایجاد پیوندی به صفحه‌ای دیگر که آدرس اینترنتی آن `url` میباشد بکار رفته است. برای تعیین مقصد و یا آدرس صفحه جدید از شناسه‌ای به نام `href` که همان `reference hyperlink` میباشد، استفاده میشود و مقدار این شناسه در واقع همان آدرس اینترنتی صفحه مقصد است. بخش قابل رویت پیوند و در واقع متنی (و یا تصویری) که توسط مرورگر نمایش داده خواهد شد و بازدیدکننده روی آن کلیک خواهد کرد میان تگ‌های `<a>` و `` قرار داده میشود و هر چند که در مثال بالا این بخش متن "Some Text" است ولی میتواند حتی یک تصویر باشد.

برای نمونه کد اجتمالی زیر پیوندی به سایت `google.com` ایجاد خواهد کرد:

```
<a href="http://www.google.com/">Visit Google Site</a>
```

و مرورگر پیوند بالا را به شکل زیر نمایش داده و در اثر کلیک روی پیوند توسط بازدیدکننده، مرورگر به سایت `Google` خواهد رفت.
Visit Google Site

شناسه target در پیوندها:

با کمک شناسه `target` امکان تعیین مقصد پیوند جدید فراهم میشود. در مثال بالا پس از کلیک روی پیوند، مرورگر سایت `google.com` را باز کرده و جایگزین سایت فعلی خواهد شد. اگر میخواهید که مرورگر پیوند را در پنجره‌ای جدید باز کند باید از شناسه `target` و مقدار `"_blank"` برای آن استفاده کنید. مثال زیر سبب باز شدن سایت `google.com` در پنجره جدیدی خواهد شد:

```
<a href="http://www.google.com/" target="_blank">Visit Google Site</a>
```

خودتان آزمایش کنید : Visit Google Site

شناسه Name :

با کمک شناسه `name` می‌توانید پیوندها را نام گذاری کنید. پیوندهای نامگذاری شده امکان حرکت میان قسمت‌های مختلف یک صفحه یا `page` را فراهم می‌کنند، مخصوصاً وقتی که محتوای یک صفحه زیاد بوده و مرورگر فقط قادر به نمایش بخشی از محتوای صفحه می‌باشد. در اینصورت مثلاً برای رفتن به آخر یک متن دیگر نیازی به `Scroll down` کردن تمامی صفحه نیست و کافی است که بازدیدکننده روی پیوندی که به آخر صفحه اشاره می‌کند کلیک کند.

مثال: برو بالای صفحه!

استفاده از پیوندهای نامگذاری شده شامل دو مرحله است:

۱- ایجاد یک پیوند نامگذاری شده (این قسمت به عنوان لنگر کار خواهد کرد):

فرم کلی یک پیوند نام گذاری شده به قرار زیر است:

`Text to be displayed`

وظیفه شناسه `name` تعیین نام برای پیوند است و مقدار آن همان نام پیوند می‌باشد. نام پیوند هم هر نام لاتینی می‌تواند باشد. در فرم کلی بالا، نام پیوند `label` و متنی که بعنوان پیوند نمایش داده می‌شود عبارت "displayed Text to be" خواهد بود. مثال زیر به ایجاد یک پیوند نامگذاری شده با نام `top` می‌پردازد:

`Here is top of my page!`

۲- ایجاد یک پیوند به پیوند نامگذاری شده دیگر:

برای دادن لینک به پیوندی نام گذاری شده، کافی است که پیوندی معمولی ایجاد کرده و در قسمت `href` آن ابتدا `url` مقصد و سپس نویسه `#` و در نهایت نام پیوند نامگذاری شده را درج کنید. مثال زیر ایجاد پیوندی است که به لینک نامگذاری شده ای به نام `top` اشاره می‌کند:

` Goto Top! `

در اثر کلیک روی پیوند بالا مرورگر مستقیماً به ابتدای بخش `top` صفحه `http://www.khaterat.com/www/index.cgi` خواهد رفت.

اگر مقصد پیوند در همان صفحه قرار دارد نیازی به قید `url` نیست و فقط نویسه `#` و سپس نام پیوند کافی است:

` Goto Top! `

چند نکته کاربردی در مورد پیوندها:

* یکی از کاربردهای رایج پیوندهای نام گذاری شده در صفحات و متونی می‌باشد که فهرست و یا لیستی از اقلام نمایش داده شده است، مانند بخش سرفصل مطالب، فهرست `FAQ` و یا ... در همین صفحه در دو مورد از پیوندهای نامگذاری شده استفاده شده است، آیا می‌توانید آنها را بیابید؟!

* اگر مرورگر نتواند یک پیوند نامگذاری شده را بیابد ابتدای صفحه مقصد را نمایش خواهد داد.

مثالهای بیشتر

باز کردن پیوند در پنجره ای جدید
این مثال نحوه ایجاد پیوندهایی که در پنجره ای جدید باز میشوند را نشان میدهد. در این حالت بازدیدکنندگان سایتتان را از دست نخواهید داد.

ارجاع به قسمت دیگری از صفحه
با کمک این مثال نحوه تعریف پیوندهای نام گذاری شده را فراگرفته و چگونگی ارجاع به قسمتهای مختلف یک صفحه اچتمل را فرا خواهید گرفت.

رهائی از شر فریم ها
این مثال کاربرد شناسه target و مقدار _top و نحوه رهائی از فریم ها را شرح خواهد داد.

نحوه ایجاد یک پیوند mailto
این مثال نحوه ایجاد یک پیوند از نوع mailto برای باز کردن برنامه مدیریت ایمیل ها را نشان میدهد. در این حالت اگر بازدیدکننده سایت برنامه های مربوط به ارسال و مدیریت ایمیل ها را نصب کرده باشد میتواند به ارسال ایمیل پردازد.

مثالی کاملتر در مورد mailto
مثال از ارسال ایمیل با امکان تعیین مقادیر cc, bcc, subject و body

Start Tag	Purpose	Anchor عنصر کاربرد
<a>	Defines an anchor	تعریف یک پیوند یا Anchor در یک صفحه اچتمل

Target Attributes	کاربرد حالت‌های مختلف شناسه target
target="_blank"	مرورگر پیوند را در یک پنجره جدید باز میکند.
target="_self"	مرورگر پیوند را همان پنجره باز میکند. (حالت پیش فرض یا default)
target="_parent"	مرورگر پیوند را فریم parent باز میکند. (کاربرد در مبحث فریمها)
target="_top"	مرورگر پیوند را در فریم مادر و اصلی باز میکند. (روشی خوب برای نجات از شر فریمها)

برای مطالعه بیشتر در مورد شناسه target سری به این صفحه بزنید.

جداول (Tables)

نمایش و چیدن اطلاعاتی که بصورت سطری و ستونی هستند مانند جدولی از اعداد و آمار تنها با استفاده از تگ های
 و <p> بسیار مشکل میباشد و به همین دلیل زبان اچتمل تگهای خاصی را برای ساختن اطلاعاتی که ذاتا جدولی (سطری و ستونی) هستند در نظر گرفته است. در این درس با المان جدول (Table) ، سلولها یا خانه ها، مرز و border در جداول و تگ های مربوطه مخصوصا تگهای

Learn HTML and CSS with w3schools | 2012

TD , TR , TABLE آشنا خواهید شد. توجه داشته باشید که جداول یکی از مهمترین المانهای چیدمان و layout میباشند.

مثالها:

جداول، آرایه ها
مثال ساده ای برای ایجاد جداول

مرز جداول
نقش شناسه border در تعیین مرز جداول

مثالهای بیشتر

جدولها

برای تعریف جداول از تگ <table> استفاده میشود. یک جدول از یک یا چند سطر که با کمک تگ <tr> تعریف میشوند، تشکیل میشود. هر ردیف یا row از یک یا چند سلول، خانه یا cell تشکیل گردیده که با کمک تگ <td> ایجاد میشوند. نام های td و tr به ترتیب خلاصه شده row table و data table میباشند.

محتوی یک سلول میتواند متن، تصویر، فهرستها، جداول دیگر، پاراگرافها و ... باشد.

مثال زیر جدولی است با دو سطر و سه ستون :

کد اجتملى جدولی با دو سطر و سه ستون

```
<table border="1" dir="rtl">
<tr>
<td>ردیف ۱، سلول ۱</td>
<td>ردیف ۱، سلول ۲</td>
<td>ردیف ۱، سلول ۳</td>
</tr>
<tr>
<td>ردیف ۲، سلول ۱</td>
<td>ردیف ۲، سلول ۲</td>
<td>ردیف ۲، سلول ۳</td>
</tr>
```

نمایش جدول روبرو توسط مرورگر

ردیف ۱، سلول ۱	ردیف ۱، سلول ۲	ردیف ۱، سلول ۳
ردیف ۲، سلول ۱	ردیف ۲، سلول ۲	ردیف ۲، سلول ۳

</table>

جدولها و شناسه border و dir :

● در مثال بالا شناسه border مرز جدول را مشخص میکند، مقدار ۱ مرزی با ضخامت یک پیکسل را نمایش خواهد داد و مقدار صفر جدول را بدون مرز نمایش خواهد داد. توجه داشته باشید که حالت پیش فرض یعنی جدولی بدون شناسه border ، جداول بدون مرز را نمایش خواهد داد.

● در مثال بالا شناسه dir یا direction و مقدار rtl برای آن، سبب تعیین نمایش جهت متون از راست به چپ خواهد شد و چون در داخل تک table تعریف شده است این خاصیت به تمامی سلولهای جدول منتقل خواهد شد.

● اگر جدول حاوی متون لاتین است نیازی به قید dir=ltr نیست و به صورت پیش فرض اطلاعات از چپ به راست چیده خواهند شد.

● ساده ترین جدول ممکن در اچتمل، جدولی است با یک سطر و یک ستون! (<table><tr><td>Hi</td></tr></table>)

سرستون در جداول (Headings)

سرستونها در جداول با کمک تگ <th> تعریف میشوند. مثال زیر نحوه تعریف سرستونها را در جداول نمایش میدهد:

کد اچتمل جدولی با سه سطر و سه ستون

نمایش جدول روبرو توسط مرورگر

```
<table border="1" dir="rtl">
```

```
<tr>
<th>سرستون ۱</th>
<th>سرستون ۲</th>
<th>سرستون ۳</th>
</tr>
```

```
<tr>
<td>ردیف ۱، سلول ۱</td>
<td>ردیف ۱، سلول ۲</td>
<td>ردیف ۱، سلول ۳</td>
</tr>
```

```
<tr>
<td>ردیف ۲، سلول ۱</td>
<td>ردیف ۲، سلول ۲</td>
<td>ردیف ۲، سلول ۳</td>
```

سرستون ۱	سرستون ۲	سرستون ۳
ردیف ۱، سلول ۱	ردیف ۱، سلول ۲	ردیف ۱، سلول ۳
ردیف ۲، سلول ۱	ردیف ۲، سلول ۲	ردیف ۲، سلول ۳

```
</tr>
</table>
```

همانطوریکه مشاهده میکنید سر ستونها مانند سلولهای معمولی جداول تعریف میشوند و فقط به جای **td** از **th** استفاده شده و لی در نمایش محتوای سر ستونها **bold** یا توپر نمایش داده خواهند شد.

خانه های خالی در جداول (Empty Cells)

اگر محتوای خانه ای از یک جدول خالی باشد چه چیزی باید نمایش داده شود؟ بسته به مرورگری که استفاده میکنید نتیجه نمایش متفاوت است و مثلا در مثال زیر خانه خالی جدول بدون مرز نمایش داده خواهد شد:

کد اچتمل جدولی با دو سطر و دو ستون

نمایش جدول روبرو توسط مرورگر

```
<table border="1" dir="rtl">
<tr>
<td>ردیف ۱، سلول ۱</td>
<td>ردیف ۱، سلول ۲</td>
</tr>
<tr>
<td>ردیف ۲، سلول ۱</td>
<td></td>
</tr>
</table>
```

ردیف ۱، سلول ۱	ردیف ۱، سلول ۲
ردیف ۲، سلول ۱	

در اینگونه موارد برای رفع مشکل کافی است که از **non-breaking space ()** یا همان نویسه و کاراکتر قاصله و بلانک استفاده شود و در مورد مثال ذکر شده مرورگر جدول را به شکل زیر نمایش خواهد داد:

کد اچتمل جدولی با دو سطر و دو ستون

نمایش جدول روبرو توسط مرورگر

```
<table border="1" dir="rtl">
<tr>
<td>ردیف ۱، سلول ۱</td>
<td>ردیف ۱، سلول ۲</td>
</tr>
<tr>
<td>ردیف ۲، سلول ۱</td>
<td>&nbsp;</td>
</tr>
</table>
```

ردیف ۱، سلول ۱	ردیف ۱، سلول ۲
ردیف ۲، سلول ۱	

توجه داشته باشید که اینبار مرزهای خانه خالی جدول بدرستی ترسیم گردیده اند.

چند نکته اساسی در مورد جداول:

- در جدول، ردیفها، خانه ها و سر ستونها از تعداد زیادی از شناسه ها میتوانيد استفاده کنید که شرح همگی آنها از حوصله این دوره مقدماتی خارج است و توصیه میشود که از مثالهای زیر و جدول واقع در انتهای این صفحه برای مطالعه هر چه بیشتر در مورد جزئیات شناسه ها استفاده شود.
- شناسه `dir` قابل استفاده در بسیاری از تگها (مثلا `table`، `tr`، `td`، `th`) میباشد. با کمک شناسه `dir` و مقدار `rtl` برای آن، جهت نمایش متون فارسی از "راست به چپ" تعیین میگردد. توجه داشته باشید که مقدار شناسه `dir` به صورت موروثی از `table` به `tr` و `th` و از `tr` و `th` به `td` خواهد رسید. مثلا برای تعیین مقدار `rtl` برای تمامی خانه های یک جدول کافی است که فقط شناسه مزبور را در تگ `table` قید کنید و نیازی به قید آن در تمامی خانه های جدول نیست. البته میتوانيد که در مورد یک ردیف خاص و یا یک خانه خاص شناسه `dir` را موردی تعیین کنید.

مثالهای بیشتر

مرز در جداول

این مثال کاربرد شناسه `border` را در نمایش جداول نشان میدهد.

تعیین جهت نمایش متن در خانه های یک جدول

این مثال کاربرد شناسه `dir` است در تعیین جهت نمایش متن در خانه های یک جدول.

عنوان جداول و تگ `caption`

مثالی در مورد تگ `caption` در تعریف جداول

توسعه سطری یا ستونی در جداول

مثالی در مورد شناسه های `colspan` و `rowspan` و ادغام خانه های جداول به صورت سطری یا ستونی

جدولی با محتویات مختلف

محتوی یک خانه جدول تقریباً میتواند هر عنصر اچتملی باشد.

فاصله گذاری بین خانه های جداول

این مثال نحوه ایجاد فاصله بین خانه های جدول را نشان میدهد. (`cellspacing`)

لایه گذاری خانه های جدول

این مثال نحوه ایجاد فاصله خالی بین خانه های جدول و مرز آنها را نشان میدهد. (cellpadding)

تعیین رنگ زمینه یا تصویر زمینه کل یک جدول

تعیین رنگ زمینه یا تصویر زمینه سراسری جدول با کمک شناسه های bgcolor و background

تعیین رنگ زمینه یا تصویر زمینه یک یا چند خانه از جدول

تعیین رنگ زمینه یا تصویر زمینه در خانه های جدول با کمک شناسه های bgcolor و background

تراز بندی محتوا در خانه های جدول

این مثال نحوه تراز بندی محتوای خانه های جدول با کمک شناسه "align" را نشان میدهد.

شناسه frame در جدول

مثالی در مورد کاربرد شناسه "frame" در المان table جهت تعیین شکل مرز جدول.

تراز بندی محتوا در خانه های جدول

این مثال نحوه تراز بندی محتوای خانه های جدول با کمک شناسه "align" و "valign" را نشان میدهد.

... فقط تصور کن ...

مثالی کاملتر و استفاده از شناسه dir در جدول

تگهای جدول

Start Tag	Purpose	کاربرد
<table>	Defines a table	تعریف جدول
<th>	Defines a table header	تعریف سرستون در جدول
<tr>	Defines a table row	تعریف ردیف ها در جدول
<td>	Defines a table cell	تعریف سلول یا خانه های یک جدول
<caption>	Defines a table caption	تعریف عنوان جدول

فریمها (Frames)

با کمک فریمها قادر به نمایش همزمان بیش از یک صفحه وب در پنجره مرورگر خواهید شد. در این درس با مفاهیم فریم (Frame) و تگ های مربوطه مخصوصا تگهای Frame و Frameset آشنا خواهید شد.

مثالها:

مجموعه فریمهای عمودی

مثالی در مورد ایجاد مجموعه فریمهای عمودی توسط تگ های frame و frameset

مجموعه فریمهای افقی

مثالی در مورد ایجاد مجموعه فریمهای افقی توسط تگ های frame و frameset

مثالهای بیشتر

فریمها (Frames)

با کمک فریمها قادر به نمایش همزمان بیش از یک صفحه وب در پنجره مرورگر خواهید شد. هر صفحه یا دریچه، فریم (frame) نامیده شده و فریمها کاملا مستقل از هم میباشند. محتوای هر فریم هم میتواند هر صفحه یا وب سایتی باشد. هر چند که فریمها طراحی سایتهای کوچک و متوسط را بسیار آسان میکنند ولی استفاده از آنها در کارهای حرفه ای توصیه نمیشود.(به جز در موارد خاص)

معایب استفاده از فریمها:

- موتورهای جستجو با صفحات طراحی شده با فریمها مشکل دارند.
- چاپ و print صفحاتی که از فریمها استفاده میکنند مشکل است.
- صفحاتی که از فریمها استفاده میکنند معمولا مبتدی بودن طراح سایت را نشان میدهد!!

اگر میخواهید صفحات نسبتا دینامیک ساخته و با هیچ زبان اسکریپتی مانند PHP یا ASP آشنائی ندارید فریمها کمک خوبی خواهند بود. همچنین فریمها برای ساختن سریع قسمتهائی از سایت که عموم به آن دسترسی ندارند مانند قسمت مدیریت (Admin) مناسب به نظر میرسند.

تگ فریم ست (Frameset)

- frameset یعنی یک Set از frame ها یا مجموعه ای از چهار چوبها
- تگ اصلی <frameset> به تعریف فریمها و تقسیم بندی پنجره مرورگر میپردازد.

- هر تگ `<frameset>` مجموعه ای از ردیفها یا ستونها را تعریف میکند.
- داخل هر فریم میتواند `frameset` دیگری قرار دهید.
- با کمک شناسه های `rows` و `cols` ابعاد و نسبت هر فریم تعیین میگردد.
- تگ `<frameset>` درون تگهای `<html>` و `</html>` قرار میگیرد.(نیازی به قید تگ `body` هنگام معرفی یک `frameset` نیست.)

تگ فریم (Frame)

- کاربرد اصلی تگ `<frame>` تعیین `source` یا نام صفحه ای است که باید در فریم ها نمایش داده شود.

مثال زیر مجموعه ای از فریمها را که شامل دو فریم عمودی میباشد معرفی میکند. از سمت چپ فریم اول ۷۵ درصد و فریم دوم ۲۵ درصد از عرض پنجره مرورگر را به خود نمایش خواهند داد. در ادامه تعریف محتویات هر فریم، درون فریم اول فایل `"frame_left.html"` و درون فریم دوم فایل `"frame_right.html"` نمایش داده خواهد شد. توجه داشته باشید که تمامی سه فایل اجتملی مربوط به این مثال در یک دایرکتوری قرار دارند.

```
<html>
<frameset cols="75%,25%">
  <frame src="frame_left.html">
  <frame src="frame_right.html">
</frameset>
</html>
```

برای نمایش مثال بالا به سه فایل نیاز دارید، `frame_right.html`، `frame_left.html` و فایل کد اجتملی بالا.

شناسه های `rows` و `cols` در تگ `frameset`

کارکرد	شناسه	مقدار شناسه
تعیین تعداد و اندازه ستونها در فریم ست	<code>cols</code>	عرض پنجره بر اساس پیکسل، درصد یا علامت*
تعیین تعداد و اندازه ردیفها در فریم ست	<code>rows</code>	عرض پنجره بر اساس پیکسل، درصد یا علامت*

مثالهای زیر را در نظر بگیرید:

مثال ۱	مثال ۲
<pre><html> <frameset cols="*,200"> <frame src="frame_2.html"> <frame src="frame_1.html"></pre>	<pre><html> <frameset cols="25%,50%,25%"> <frame src="frame_3.html"> <frame src="frame_2.html"></pre>

<code></frameset> </html></code>	<code><frame src="frame_1.html"> </frameset> </html></code>
نمایش توسط مرورگر	نمایش توسط مرورگر

در مثال یک عرض فریم اول از سمت راست ۲۰۰ پیکسل تعیین شده و بقیه عرض پنجره که با علامت * مشخص میشود به فریم دوم اختصاص خواهد یافت.

در مثال دوم عرض پنجره با مقادیر درصدی به فریمها اختصاص یافته است. در مورد شناسه rows دقیقا قراردادهای بالا برقرارند.

نکات کاربردی:

* اگر فریمهای طراحی شده مرزهای قابل مشاهده دارند توسط کاربر قابل تغییر اندازه (resize) خواهند بود و برای جلوگیری از تغییر اندازه فریمها توسط کاربر باید از شناسه ای به نام noresize در داخل تگ <frame> استفاده شود. در این مثال noresize نمیتوانید که فریمها را تغییر اندازه بدهید.

* اگر میخواهید که فریمها scroll نشوند (حرکت میان محتویات صفحه توسط ماوس) باید در تگ frame مقدار شناسه scrolling را برابر "no" قرار دهید. (مقادیر ممکن "no, yes" و "auto" میباشد). در این مثال scrolling صفحات قابل scroll نیستند.

* اگر میخواهید که مرز بین فریمها نمایش داده نشوند باید در تگ frame مقدار شناسه frameborder را برابر "۰" قرار دهید. (مثال frameborder)

* درصد بسیار کمی از مرورگرها از تگهای مربوط به frame و frameset پشتیبانی نمیکند. برای این موارد باید از تگ <noframes> جهت نمایش صفحه جایگزین استفاده شود.

* بعضی از مرورگرها به اولین نویسه و شروع تعریف یک فایل frameset حساس هستند، مطمئن باشید که هیچ کاراکتر اضافی مانند space یا new link قبل از tag frameset در ابتدای صفحه وجود ندارد.

مثالهای بیشتر

مجموعه فریمهای مرکب

مثالی در مورد ایجاد مجموعه ای مرکب از فریمهای عمودی و افقی

مثالی کاربردی در مورد فریمها

در این مثال کاربردی با کمک سه فریم امکان نمایش صفحات مختلف به آسانی فراهم میشود. فریم افقی و بالائی سرتیتر و نام اصلی سایت را نمایش خواهد داد و فریم سمت راست لینکهای مهم صفحه را در بر گرفته و امکان navigation را فراهم میسازد و نهایتا فریم اصلی و

سمت چپ صفحات مختلف سایت را به نمایش خواهد گذاشت. توجه داشته باشید که برای این مثال حداقل به یک فایل برای معرفی مجموعه فریمها (frameset) و سه صفحه اچتمل برای فریمهای بالائی، سمت چپ و راست احتیاج خواهد بود.

فریمهای شناور و یا درجا (Inline frame)

مثالی در مورد فریمهای شناور و درجا (iframe)، فریمی درون یک صفحه اچتمل

مثالی کاربردی در مورد فریمهای درجا (Inline frame)

مثالی کاربردی در مورد فریمهای شناور و درجا (iframe) با کمک ۷ صفحه اچتمل مختلف

شوخی با فریم های شناور

مثالی در مورد نمایش درایو لوکال (مثلا C:\ بازدیدکنندگان سایت به آنها

فریمها و پیوندهای نامگذاری شده

مثالی کامل در مورد فریمها و پیوندهای نامگذاری شده و نحوه استفاده از آنها در طراحی سایتی ساده متشکل از چهار فایل اچتمل

فریمها و پیوندهای نامگذاری شده کامل

همان مثال بالا در صفحه کامل مرورگر با شناسه های کامل در تگ frame

تگ های فریم

Start Tag	Purpose	کاربرد
<frameset>	Defines a set of frames	تعریف مجموعه ای از فریم ها
<frame>	Defines a sub window (a frame)	تعریف یک فریم
<noframes>	Defines a noframe section for browsers that do not handle frames	تعریف جایگزین برای مرورگرهایی که از فریم ها پشتیبانی نمیکنند.
<iframe>	Defines an inline sub window (frame)	تعریف فریم های درجا (inline)

فهرستها (Lists)

این درس شما را با نحوه نمایش انواع فهرستها شامل فهرستهای مرتب (ordered) ، نامرتب (unordered) و فهرستهای تعریفی (definition lists) و تگ های مربوطه یعنی li, ol, ul, dl, dt و dd آشنا خواهد کرد.

مثالها:

فهرستهای نامرتب

مثالی بسیار ساده در مورد فهرستهای نامرتب

فهرستهای مرتب

مثالی در مورد فهرستهای مرتب و نحوه نمایش متون فارسی

مثالهای بیشتر

فهرستهای نامرتب (unordered list)

فهرستهایی هستند از یک یا چند قلم اطلاعات که معمولاً با دایره ای کوچک و سیاه رنگ در شروع نمایش داده میشوند. یک فهرست از نوع نامرتب با تگ `` شروع شده و هر رقم اطلاعات با تگ `` مشخص میگردد.

نمایش توسط مرورگر

- Red
- Blue
- Green

HTML code

```
<ul>
<li>Red</li>
<li>Blue</li>
<li>Green</li>
</ul>
<ul dir="rtl" >
<li>قهوه</li>
<li>شیر</li>
<li>چای</li>
</ul>
```

داخل تگهای `` میتوانید از پاراگرافها، تگهای `br`، تصاویر و حتی فهرستهای دیگر استفاده کنید. به شناسه `dir` در مثال بالا توجه کنید، امکان تعیین سمت و جهت نمایش متون در تمامی تگهای مربوط به فهرستها ممکن میباشد.

فهرستهای مرتب (ordered list)

این نوع فهرست بسیار شبیه فهرستهای نامرتب بوده و فقط مرورگر به هنگام نمایش اقلام اطلاعاتی از اعداد به جای دایره های کوچک سیاه رنگ استفاده میکند. یک فهرست از نوع مرتب با تگ `` شروع شده و همانند فهرستهای نامرتب هر رقم اطلاعات با تگ `` مشخص میگردد.

نمایش توسط مرورگر

1. Red
2. Blue
3. Green

HTML code

```
<ol>
<li>Red</li>
<li>Blue</li>
<li>Green</li>
</ol>
```

```

.۱ قهوه <ol dir="rtl" >
      <li>قهوه</li>
.۲ شیر <li>شیر</li>
.۳ چای <li>چای</li>
      </ol>

```

داخل تگهای `` میتوانید از پاراگرافها، تگهای `br`، تصاویر و حتی فهرستهای دیگر استفاده کنید. به شناسه `dir` در مثال بالا توجه کنید، امکان تعیین سمت و جهت نمایش متون در تمامی تگهای مربوط به فهرستها ممکن میباشد.

فهرستهای تعریفی (definition list)

فهرست تعریفی فهرستی است از اصطلاح ها و تشریح و توصیف آنها.

فهرستهای تعریفی با `<dl>` شروع شده و هر زوج اصطلاح و تعریف مربوط به آن با تگهای `<dt>` و `<dd>` تعریف میشوند.

نمایش توسط مرورگر	HTML code
HTTP	<code><dl></code>
Hypertext Transfer Protocol	<code><dt>HTTP</dt></code>
FTP	<code><dd>Hypertext Transfer Protocol</dd></code>
File Transfer Protocol	<code><dt>FTP</dt></code>
IP	<code><dd>File Transfer Protocol</dd></code>
Internet Protocol	<code><dt>IP</dt></code>
	<code><dd>Internet Protocol</dd></code>
	<code></dl></code>
مقدمه	<code><dl dir="rtl" ></code>
خلاصه ای از مقدمه	<code><dt >مقدمه</dt></code>
فصل ۱	<code><dd>خلاصه ای از مقدمه</dd></code>
خلاصه ای از فصل ۱	<code><dt>فصل ۲</dt></code>
فصل ۲	<code><dd>خلاصه ای از فصل ۱</dd></code>
خلاصه ای از فصل ۲	<code><dt>فصل ۲</dt></code>
	<code><dd>خلاصه ای از فصل ۲</dd></code>
	<code></dl></code>

داخل تگهای تشریح یعنی `<dd>` میتوانید از پاراگرافها، تگهای `br`، تصاویر و حتی فهرستهای دیگر استفاده کنید. به شناسه `dir` در مثال بالا توجه کنید، امکان تعیین سمت و جهت نمایش متون در تمامی تگهای مربوط به فهرستها ممکن میباشد.

مثالهای بیشتر

انواع فهرستهای مرتب

مثالی در مورد فهرستهای مرتب به همراه استفاده از شناسه `type` جهت تعیین نوع نمایش فهرست

انواع فهرستهای نامرتب

مثالی در مورد فهرستهای نامرتب به همراه استفاده از شناسه `type` جهت تعیین نوع نمایش فهرست

فهرستهای تودرتو

مثالی در مورد فهرستهای ترکیبی. اعضای فهرست میتوانند از فهرستهای دیگر تشکیل شوند.

فهرستهای تعریفی

مثالی در مورد فهرستهای تعریفی به همراه استفاده از شناسه `dir`

تگهای فهرست

Start Tag	Purpose	کاربرد
<code></code>	Defines an ordered list	تعریف فهرستهای مرتب
<code></code>	Defines an unordered list	تعریف فهرستهای نامرتب
<code></code>	Defines a list item	تعریف یک آیتم و قلم از یک فهرست
<code><dl></code>	Defines a definition list	تعریف فهرستهای تعریفی
<code><dt></code>	Defines a definition term	تعریف اصطلاح در فهرستهای تعریفی
<code><dd></code>	Defines a definition description	تعریف معنی و شرح اصطلاح در فهرستهای تعریفی

فرمها (Forms)

از طریق استفاده از عنصر `<form>` و چند تگ مرتبط قادر به دریافت اطلاعات از بازدیدکنندگان صفحاتتان و یا تبادل اطلاعات بین صفحات مختلف خواهید شد. با کمک این گروه از تگ ها قادر به نمایش باکسهای ورود اطلاعات متن (text fields) ، چک باکسها (-check boxes) ، رادیو باتونها (radio-buttons) و ... شده و همچنین امکان گذاشتن دکمه های ارسال (button submit) و یا reset را خواهید داشت. این فصل با شرح تگ های `form` و `input` و ... به چگونگی ایجاد ارتباط با بازدیدکنندگان و گرفتن اطلاعات از آنها بصورت لاتین و یا فارسی خواهد پرداخت.

توجه داشته باشید که برای پردازش اطلاعات دریافتی از کاربر باید با یکی از زبانهای `cgi` از قبیل `ASP` ، `perl` ، `PHP` ، `CFM` ، `JSP` یا `Java` آشنائی داشته باشید. مثالهای `cgi` مدرسه وب از زبان اسکریپت `Perl` یا `PHP` استفاده میکنند.

درس قبلی، درس بعدی

مثالها:

ورودیهای متن یا Text fields

مثالی ساده در مورد ایجاد text field ها . با کمک text field ها کاربران قادر به وارد کردن و تایپ اطلاعات متنی خواهند شد.

ورودیهای رمز عبور یا Password fields

مثالی ساده در مورد ایجاد password field ها . با کمک password field ها کاربران قادر به ورود رمز عبور یا پسورد خواهند شد، مرورگرها برای امنیت اطلاعات رمزهای عبور را رمزگذاری میکنند و معمولا از '#' برای نمایش اطلاعات استفاده میکنند.

مثالهای بیشتر

فرمها (Forms)

تمامی عناصر و تگهایی که تاکنون دیده اید فقط به نمایش اطلاعات پرداخته اند و هیچکدام به گرفتن و اخذ اطلاعات از کاربر نپرداخته اند. فرمها که با تگ <form> معرفی میشوند، طراح سایت را قادر به جمع آوری و اخذ اطلاعات از بازدیدکننده سایت خواهند کرد. عنصر فرم و گروهی از عناصر و تگهای درون آن به دریافت و ارسال اطلاعات به سمت وب سرور کمک خواهند پرداخت. (تمامی تگهای مربوط به فرمها درون تگهای <form> و </form> قرار میگیرند.)

درون تگ فرم گروه زیادی از عناصر و تگهای مختلف قابل استفاده هستند از قبیل عناصر ورود متن یک سطری (text fields)، عناصر ورود متنهای چند سطری (Textarea)، منوهای drop-down و radio buttons و منوهای drop-down و غیره.

تگ Input

به عنوان پر مصرف ترین تگ مربوط به ورود اطلاعات باید از تگ <input> نام برد. در این تگ شناسه ای به نام type به تعیین نوع اطلاعات ورودی اختصاص دارد. مقادیر ممکن برای این شناسه به قرار زیرند:
text , checkbox , radio , password , hidden , submit , reset , button , file , image
در ادامه به شرح بعضی از type های کاربردی خواهیم پرداخت:

ورودیهای متن (Text Fields)

اگر میخواهید که بازدیدکننده اطلاعاتی از قبیل متن، اعداد و ... را وارد کند از شناسه ای با مقدار "text" استفاده میشود.

نمایش توسط مرورگر

First name:

کد اچتمل

```
<form>
First name:
<input type="text" name="firstname">
```

Last name: <input type="text"/>	
نام خانوادگی: <input type="text"/>	Last name:
نام: <input type="text"/>	<input type="text" name="lastname" >
نام خانوادگی: <input type="text"/>	</form>
	<form dir="rtl" >
	نام:
	<input type="text" name="firstname" >
	نام خانوادگی:
	<input type="text" name="lastname" >
	</form>

توجه داشته باشید که تگ <form> چیزی را به نمایش نخواهد گذاشت بلکه تگهای درون آن توسط مرورگر نمایش داده خواهند شد. در مثال فارسی بالا به شناسه dir و مقدار rtl آن توجه داشته باشید. لازم به یادآوری است که اغلب مرورگرها در حالت پیش فرض برای ورودیهای متن اندازه ۲۰ کاراکتر را در نظر میگیرند و اگر میخواهید که اندازه پیش فرض ورودیهای متن را تغییر دهید باید از شناسه ای به نام size استفاده کنید.

ورودیهای Radio Buttons

اگر میخواهید که بازدیدکننده فقط یک گزینه را از بین چند گزینه محدود انتخاب کند، از مقدار "radio" برای شناسه type استفاده کنید:

نمایش توسط مرورگر	کد اجتم
<input type="checkbox"/> Male	<form>
<input type="checkbox"/> Female	<input type="radio" name="sex" value="male" > Male
	<input type="radio" name="sex" value="female" > Female
	</form>
<input type="checkbox"/> مرد	<form dir="rtl" >
<input type="checkbox"/> زن	<input type="radio" name="sex" value="male" > مرد
	<input type="radio" name="sex" value="female" > زن
	</form>

همانطور که مشاهده میشود فقط امکان یکی از گزینه ها برای کاربر میسر است.

ورودیهای Checkboxes

اگر میخواهید که بازدیدکننده یک یا چند گزینه را از بین چندین گزینه محدود انتخاب کند، از مقدار "checkbox" برای شناسه type استفاده کنید:

نمایش توسط مرورگر	کد اجتم
<input type="checkbox"/> I have a bike	<form>
	<input type="checkbox" name="bike" >
	I have a bike

<input type="checkbox"/>	I have a car	<pre>
 <input type="checkbox" name="car" > I have a car </form> <form dir="rtl"> <input type="checkbox" name="bike" > دوچرخه دارم
 <input type="checkbox" name="car" > ماشین دارم </form></pre>
<input type="checkbox"/>	دوچرخه دارم	
<input type="checkbox"/>	ماشین دارم	

شناسه Action و دکمه Submit در فرمها (Submit Button & Form's Action Attribute)

در فرمها برای ارسال اطلاعات کسب شده از دکمه ای به نام ارسال یا Button Submit استفاده میشود و در اثر کلیک کاربر بروی این دکمه "Submit" ، اطلاعات درون فرم توسط مرورگر به صفحه ای که شما تعیین میکنید ارسال خواهند شد. برای تعیین مقصد ارسال اطلاعات باید درون تگ form از شناسه ای به نام action استفاده کنید. مقدار شناسه action آدرس یا url صفحه ای است که به دریافت و سپس پردازش اطلاعات دریافتی خواهد پرداخت. معمولا فایل‌های بخش action برنامه ها و اسکریپت هائی نوشته شده با زبانهای cgi مانند PHP ، Perl ، ASP و ... بوده و وظیفه آنها دریافت اطلاعات فرمها و سپس پردازش آنها میباشد.

تعیین مقدار "submit" برای شناسه type سبب نمایش دکمه Submit یا ارسال خواهد گردید.

نمایش توسط مرورگر	کد اچتمل
Username: <input type="text"/>	<pre><form name="input" action="form_action.php"> Username : <input type="text" name="user"> <input type="submit" value="Submit"> </form></pre>
نام کاربر <input type="text"/>	<pre><form name="input" action="form_action.php" dir="rtl"> نام کاربر <input type="text" name="user"> <input type="submit" value="Submit"> </form></pre>

در مثال بالا در باکس ورودی متن، کلمه ای را وارد کرده و دکمه ارسال را کلیک کنید. مرورگر به محض کلیک دکمه ارسال، اطلاعات درون فرم را که در این مثال متنی ساده است به سمت آدرسی (URL) که در شناسه action تعیین شده است ارسال کرده و در این مثال برنامه ای به زبان PHP با نام form_action.php در سمت سرور به دریافت و پردازش اطلاعات دریافتی خواهد پرداخت.

به یاد داشته باشید که آدرس قید شده در بخش action میتواند آدرس اسکریپتی در همان سایت و یا سایت دیگری باشد، همچنین میتواند آدرسی نسبی یا مطلق باشد.

ورود متن فارسی در عناصری مانند Text field یا Text area:

همانطور که در مثالهای بالا مشاهده کردید در مورد المانهای Text field و Textarea مانند بسیاری دیگر از المانها با کمک شناسه dir میتوان سمت و جهت ورود اطلاعات را "از راست به چپ" تعیین کنید. ولی اگر کاربری ویندوز فارسی نداشته باشد، از کامپیوترهای مکیتاش استفاده کند و یا سیستمش را فارسی نکرده باشد قادر به ورود اطلاعات به زبان فارسی نخواهد بود و نویسه های تایپ شده لاتین خواهند بود! در اینگونه موارد چه باید کرد؟

معمولا برنامه نویسان وب برای فراهم ساختن امکان ورود متن فارسی از یکی از دو روش زیر استفاده میکنند:

- استفاده از اپلتهای جاوا یا استفاده از فلش
- استفاده از زبان جاوا اسکریپت

مثالهای بیشتر

حالتهای مختلف ورودی Input

در تگ input و با کمک شناسه type و مقادیر مختلف برای آن میتوان انواع مختلف اطلاعات را از وبگرد گرفت.

Checkboxes

مثالی در مورد نحوه تعریف فرمی متشکل از Checkbox ها

Radiobuttons

مثالی در مورد نحوه تعریف فرمی متشکل از Radio button ها

فهرست کرکره ای ساده

مثالی در مورد نحوه تعریف فرمی متشکل از drop down box ها یا فهرستهای کرکره ای

فهرست کرکره ای دیگر

مثالی در مورد نحوه تعریف فرمی متشکل از drop down box ها با تعیین پیش گزینه

Textarea

مثالی در مورد نحوه تعریف فرمی متشکل از Textarea ، ناحیه ای برای ورود بیش از یک سطر متن

تعریف فرمی با کمک دکمه یا button

مثالی در مورد نحوه تعریف فرمی با کمک دکمه یا button

ترسیم مرز در اطراف فرم

مثالی در مورد نحوه تعریف فرمی دلخواه و ترسیم مرزی بدور آن به همراه تعیین عنوانی برای فرم با کمک عنصر Fieldset

ارسال ایمیل با کمک فرمها

مثالی در مورد نحوه ارسال ایمیل با کمک فرمها

ساختن فهرستی از سایتهای مورد علاقه

مثالی در مورد نحوه استفاده از تگ فرم و منوهای کرکره ای و کمی جاوا اسکریپت برای ساخت فهرستی از سایتهای

تگهای فرم

Start Tag	Purpose	کاربرد
<form>	Defines a form for user input	تعریف فرم ورود اطلاعات
<input>	Defines an input field	تعریف ورودی از نوع Input
<textarea>	Defines a text-area (a multi-line text input control)	تعریف ورودی متن چند سطری یا text-area
<label>	Defines a label to a control	تعریف برچسب یا label
<fieldset>	Defines a fieldset	تعریف fieldset
<legend>	Defines a caption for a fieldset	تعریف عنوان برای fieldset ها
<select>	Defines a selectable list (a drop-down box)	تعریف فهرستهای انتخابی یا drop-down box
<optgroup>	Defines an option group	تعریف option group ها
<option>	Defines an option in the drop-down box	تعریف گزینه ای از drop-down box ها
<button>	Defines a push button	تعریف دکمه فشاری، متفاوت با button تگ input

تصاویر (Images)

در این درس با تگ `img` ، شناسه `src` و `alt` ، تگهای `Map` و `Area` و چگونگی نمایش تصاویر و همچنین تنظیم محل آنها در سندهای اچتمل آشنا خواهید شد.

مثالها:

درج تصویر

نحوه اضافه کردن تصویر به صفحات اچتمل با کمک تگ `img`

درج تصویری واقع در فضای آدرسی دیگر

درج تصویری واقع در دایرکتوری غیر از آدرس صفحه اچتمل و یا تصویری واقع در فضای آدرسی دیگر سایتهای

مثالهای بیشتر

تگ `img` و شناسه `src` یا `source`

در زبان اچتمل تصاویر را با کمک تگ `` تعریف میکنند. تگ `` از نوع تگهای خالی است، بدین معنا که فقط دارای یک یا چند شناسه و `attribute` بوده و دارای تگ انتهایی یا `` نیست.

مهمترین شناسه برای درج و تعریف یک تصویر `src` یا `source` نام دارد و مقدار این شناسه آدرس یا `URL` تصویر میباشد. تصاویری که مرورگر در یک صفحه وب نمایش میدهد میتوانند در همان دایرکتوری صفحه اچتمل، در دیگر دایرکتوریهای همان سایت و یا در فضای بیکران وب قرار داشته باشند.

شکل ساده درج یک تصویر:

```

```

مقدار `url` آدرس اینترنتی تصویر میباشد و مثلاً اگر تصویری که قرار است که در صفحه درج شود در همان دایرکتوری قرار دارد و نامش `test.jpg`، کافی است که به جای `url` فقط نام تصویر یعنی `test.jpg` را بنویسید و اگر تصویر در سایتی دیگر قرار دارد `url` کامل آنرا بنویسید. مثلاً تصویر لوگو سایت `google.com` دارای `url` ی برابر `http://www.google.com/images/logo.gif` است. در این `url` نام تصویر `logo.gif` بوده، در دایرکتوری `images` قرار داشته و روی وب سایت `google.com` قرار دارد.

شناسه های `width` و `height` در تگ `img`

کاربرد شناسه های `width` و `height` تعیین عرض و ارتفاع نمایش تصویر است. اگر از این شناسه ها استفاده نکنید، مرورگر ابتدا تصویر را لود کرده و سپس طول و عرض آنرا یافته و در نهایت نمایش میدهد. با نوشتن مقادیری غیر از مقادیر واقعی طول و عرض تصویر میتوانید به تغییر شکل و ابعاد آن بپردازید و آنرا مثلاً کوچکتر، بزرگتر و یا کشیده تر نمایش دهید. مثال زیر نحوه درج تصویری را با قید ابعاد آن شرح میدهد:

```

```

توصیه میشود که حتماً ابعاد تصویر را با کمک شناسه های `width` و `height` قید کنید، اینکار سبب کمک به مرورگر در تسریع شروع نمایش صفحه خواهد شد. (در واقع با قید ابعاد تصویر، مرورگر قبل از لود تصویر میداند که چه مقدار از صفحه نمایش باید به تصویر مزبور اختصاص داده شود و قبل از لود شدن تمامی تصاویر مرورگر شروع به نمایش قالب و چارچوب کلی صفحه خواهد کرد.)

شناسه `Alt` یا `"alternate text"` یا متن جایگزین

اگر مرورگر به هر دلیلی نتواند که یک تصویر را نمایش دهد و یا مرورگر از نوع مرورگرهایی باشد که فقط متن را نمایش میدهند، متن و `text`

ی که با کمک شناسه alt تعیین شده است نمایش داده خواهد شد. همچنین در اغلب مرورگرها با قرارگرفتن ماوس بروی یک تصویر، متن تعیین شده توسط شناسه alt نمایش داده خواهد شد. با کمک alt اطلاعات اضافی مربوط به تصویر قابل نمایش میباشد. مثال زیر نحوه درج تصویری را با قید متن جایگزین آن شرح میدهد:

```

```

وبلاگها و اضافه کردن تصاویر:

اگر وبلاگی داشته و میخواهید که تصویری به آن اضافه کنید دو حالت مختلف ممکن است که پیش آید:

- الف) تصویر روی سایتی دیگر قرار داشته و شما هم میخواهید از آن تصویر که در سایتی دیگر قرار دارد استفاده کنید. در این حالت از آدرس مستقیم تصویر در شناسه src استفاده کنید، مانند :

```
< width=258 height=78 "img src="http://www.google.ca/images/hp0.gif">
```

در این مثال تصویر روی سایت google.ca قرار داشته و لزومی به ذخیره آن توسط شما نیست.

- ب) تصویر روی هارد کامپیوتر شما است و هنوز بروی وب منتقل نشده است. در این حالت چون بعضی سیستم های وبلاگ امکان ذخیره تصاویر را به شما نمی دهند باید ابتدا تصویرتان را با کمک FTP یا روشهای دیگر به روی سایت یا هر وب سرور دیگری منتقل کنید. شرکتهای مختلفی فضای مجانی برای ساخت صفحات وب و ذخیره اطلاعات و تصاویر در اختیار شما قرار میدهند و در یکی از سؤالهای این FAQ فهرست این شرکتهای را میتوانید ببینید. بنابراین ابتدا فضائی در وب تهیه کرده، سپس تصویرتان را Upload کرده (با کمک FTP یا روشهای دیگر) و سپس در شناسه src از تگ img آدرس جدید تصویر را قید کنید.

چند نکته مهم:

- اگر در یک صفحه اچتمل از ۱۰ تصویر استفاده شده باشد، مرورگر باید ۱۱ فایل را لود کند. (خود صفحه به علاوه ۱۰ تصویر)
- استفاده از تصاویر، سرعت لود شدن صفحات را پایین می آورد. با احتیاط از تصاویر استفاده کنید و به اندازه و حجم تصویر توجه داشته باشید.
- لود تصویری با حجم ۵۰ کیلوبایت برای کسی که از مودمی با سرعت ۲۸kbps استفاده میکند حداقل ۱۵ ثانیه طول خواهد کشید.
- برای دیدن مشخصات تصاویر در وب کافی است که روی تصویر Right click کرده و سپس در بخش properties آدرس، ابعاد و ظرفیت تصویر را ببینید.
- برای ذخیره تصاویر در وب کافی است که روی تصویر Right click کرده و سپس در بخش "Save picture as" آنرا ذخیره کنید.

مثالهای بیشتر

تصاویر زمینه

مثالی در مورد نحوه تعیین تصاویر زمینه با کمک شناسه background در تگ body

ترازبندی تصاویر

این مثال نحوه ترازبندی تصاویر در کنار متون را نشان میدهد.

تصاویر شناور در متن

آزاد گذاشتن تصویر در سمت چپ یا راست

نمایش تصاویر در ابعاد مختلف

مثالی در مورد استفاده از شناسه های width و height و نمایش تصاویر در ابعاد مختلف

نمایش متن جایگزین برای تصاویر

مثالی در مورد استفاده از شناسه alt

استفاده از تصاویر هنگام ایجاد پیوند ها

مثالی در مورد نحوه استفاده از تصاویر هنگام ایجاد پیوند ها

ایجاد image map ها

مثالی در مورد ایجاد image map . در این مثال هر بخش از نواحی تعریف شده قابل کلیک بوده و به پیوندی اختصاصی اشاره میکند.

برنامه ای خوب و متن باز برای FTP را میتوانید اینجا دانلود کنید:

Filezilla

تگ های تصاویر

Start Tag	Purpose	کاربرد
	Defines an image	درج تصویر
<map>	Defines an image map	تعریف Image map یا ???
<area>	Defines an area inside an image map	تعریف ناحیه ای در داخل Image map

مرجع : HTML 4.01 Specification - فهرست عناصر - فهرست شناسه ها

پس زمینه (Background)

در این درس با چگونگی تعیین تصاویر و رنگ پس زمینه (Background) آشنا خواهید شد. رنگ و یا تصویر پس زمینه عامل بسیار مهمی در طراحی صفحات اچتمل بوده و استفاده از رنگ و یا تصویر مناسب نقشی تعیین کننده در خوانا بودن صفحات دارد.

مثالها:

رنگ متن و زمینه متناسب
انتخاب رنگ مناسب متون و پس زمینه به خوانا بودن متن کمک زیادی خواهد کرد.

رنگ متن و زمینه نامتناسب
مثالی از انتخاب نامناسب رنگ متن و پس زمینه

مثالهای بیشتر

رنگ و تصویر زمینه در صفحات اجتملی:

تگ `<body>` دارای دو شناسه مهم برای تعیین رنگ و تصویر پس زمینه میباشد. پس زمینه یک صفحه اجتملی، رنگ و یا یک تصویر میتواند باشد.

شناسه `bgcolor` برای رنگ پس زمینه (`color background`)

شناسه `bgcolor` رنگ پس زمینه را تعیین میکند. مقدار رنگ پس زمینه میتواند نام رنگ (مثل `red` یا `blue`) ، مقدار آن بر اساس استاندارد `RGB` و یا حتی مقدار هگزادسیمال (`Hexadecimal`) باشد.

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

در تمامی مثالهای بالا با کمک شناسه `bgcolor` رنگ زمینه صفحه "سیاه" تعیین میشود.

شناسه `Background` برای تصویر پس زمینه:

برای تعیین تصویری که به عنوان پس زمینه صفحه نمایش داده میشود از شناسه `background` استفاده میشود. مقدار این شناسه `URL` یا آدرس تصویر (`image`) میباشد. اگر ابعاد تصویر کوچکتر از ابعاد پنجره اصلی مرورگر باشد، تصویر زمینه تکثیر خواهد شد تا حدی که تمام صفحه نمایش از تصویر زمینه پوشانده شود.

```
<body background="bg.gif">
<body background="http://www.khaterat.com/images/bg.gif">
```

در مثال اول از آدرس دهی نسبی (`relative`) برای تعیین `URL` یا آدرس تصویر استفاده شده و در واقع در این مثال فایل اجتملی و تصویر

Learn HTML and CSS with w3schools | 2012

bg.gif باید در یک دایرکتوری قرار گیرند. در مقابل مثال دوم از آدرس دهی مطلق (absolute) برای تعیین آدرس تصویر استفاده شده است و همانطور که میدانید آدرس تصویر در هر نقطه از فضای وب و روی هر وب سروری میتواند باشد.

نکات مهم: اگر از تصاویر پس زمینه در صفحات اچتمل استفاده میکنید به نکات زیر توجه کنید :

- هر تصویری با فرمت png, bmp, gif و یا jpg قابل استفاده به عنوان تصویر زمینه میباشد.
- توصیه میشود از تصاویر بزرگتر از ده کیلوبایت به عنوان تصویر پس زمینه استفاده نکنید، در غیر این صورت سرعت load صفحه بشدت پایین خواهد آمد.
- از تصاویر زمینه هماهنگ با دیگر تصاویر و متون صفحه استفاده کنید.
- در رزولوشنهای مختلف به دیدن و آزمایش تصاویر پس زمینه پردازید.
- پر مصرف ترین رنگهای زمینه وب سایتهای، رنگهای سفید، سیاه و خاکستری میباشد.
- اغلب سایتهای وب از تصاویر پس زمینه استفاده نمی کنند! مثلا همین سایت!!!
- توجه داشته باشید که استفاده از شناسه های bgcolor و background مربوط به استاندارد قدیمی اچتمل میباشد.

نکته مهم در سازگاری با نسخه های جدید اچتمل و XHTML

شناسه های bgcolor و background در تگ <body> در نسخه ۴ اچتمل و XHTML پشتیبانی نمیشوند و در واقع از رده خارج محسوب میشوند. در عوض کنسرسیوم W3C استفاده از Style Sheet ها را توصیه میکند و در واقع در نسخه ۴ اچتمل و استاندارد جدید XHTML تمامی اطلاعات مربوط به نمایش یا presentation و layout باید در style sheet ها قرار گیرند.

مثالهای بیشتر

استفاده مناسب از تصویر زمینه ۱
در این مثال بدلیل استفاده از تصویر مناسب برای زمینه، نوشته های صفحه کاملا خوانا هستند.

استفاده مناسب از تصویر زمینه ۲
در این مثال در اثر استفاده از تصویر مناسب برای زمینه، نوشته های صفحه کاملا خوانا هستند.

استفاده نامناسب از تصویر زمینه
در این مثال نوشته های صفحه خوانا نیستند.

استفاده از تصویر زمینه با آدرس مطلق
در این مثال آدرس تصویر زمینه بصورت مطلق و کامل آورده شده است.

تعیین رنگ زمینه با Style Sheet ها

در این مثال چگونگی استفاده از Style Sheet ها برای تعیین رنگ زمینه آورده شده است.

قلم ها (HTML Fonts)

در این درس یاد خواهید گرفت که چگونه نوع قلم، اندازه و رنگ نوشته ها در صفحات اچتمل تعیین میشوند. با وجودیکه هنوز بعضی از اچتمل کاران از تگ `` برای تعیین نام و نوع قلم، رنگ و اندازه متون استفاده میکنند ولی در استاندارد جدید اچتمل (نسخه ۴) و همچنین XHTML از تگ `` پشتیبانی نمیشود و استفاده از Style Sheet ها به عنوان جایگزین معرفی شده است.

تگ ``

در نسخه های ۳.۲ و قبلی تر زبان اچتمل از تگ `` برای تعیین نام قلم، رنگ و اندازه متون استفاده میشود. در این تگ شناسه `face` نام قلم، شناسه `size` اندازه قلم و شناسه `color` معرف رنگ متن خواهد بود. مثالهای زیر کاربرد تگ `` خدایامرز را نشان میدهند:

کد اچتمل	نمایش کد روبرو توسط مرورگر
<pre><p> This is a paragraph. </p> <p dir="rtl"> متنی با قلم Times و رنگ FF9900 و اندازه ۱ </p> <p dir="rtl"> متنی با قلم Tahoma و رنگ قرمز و اندازه ۲ </p></pre>	<p>This is a paragraph.</p> <p style="text-align: center;">متنی با قلم Times و اندازه ۱ و رنگ FF9900</p> <p style="text-align: center;">متنی با قلم Tahoma و اندازه ۲ و رنگ قرمز</p>

شناسه های تگ ``

Attribute	Example	کاربرد
<code>size="number"</code>	<code>size="2"</code>	تعیین اندازه قلم
<code>size="+number"</code>	<code>size="+1"</code>	افزایش اندازه قلم
<code>size="-number"</code>	<code>size="-1"</code>	کاهش اندازه قلم
<code>face="face-name"</code>	<code>face="Times"</code>	تعیین نام قلم
<code>color="color-value"</code>	<code>color="#eef00"</code>	تعیین رنگ قلم
<code>color="color-name"</code>	<code>color="red"</code>	تعیین رنگ قلم

*** بیان مختصر عملکرد تگ در این دروس فقط جهت اطلاع بوده و استفاده از آنها در طراحی صفحات اجتملى توصیه نمىشود.

ديگر از تگ استفاده نكنيد!

تگ از رده خارج محسوب مىشود و كنسرسیوم جهانی تعیین استانداردهای وب (World Wide Web Consortium, W3C) این تگ را از مجموعه تگهای استاندارد اجتملى کنار گذاشته است و اكيدا توصیه مىشود كه از Style Sheet ها برای تعريف هر گونه خواص نمايشی و آرايشی (Presentation & layout) المانها در اجتملى استفاده شود.

روش صحيح تعيين نام قلم، اندازه و رنگ متون با كمك Style Sheet ها:

هر چند كه سالها است كه مبحث و كلاسهای درس Style Sheet ها در "مدرسه وب" در دست تهيه بوده و اميدواريم كه بزودی ارائه شوند! ولی تا قبل از آن استفاده از شناسه ای به نام style در بسياری از المانها كار گشا خواهد بود. مثالهای ساده زیر نحوه تعيين نام قلم، اندازه و رنگ متون را با كمك شناسه style به نمايش خواهند گذاشت:

تعيين نام قلم

مثالی در مورد تعيين نام قلم در يك استيل درجا (inline)

تعيين اندازه قلم

مثالی در مورد تعيين اندازه قلم در يك استيل درجا (inline)

تعيين رنگ قلم

مثالی در مورد تعيين رنگ قلم در يك استيل درجا (inline)

تعيين نام، اندازه و رنگ قلم

مثالی در مورد تعيين نام، اندازه و رنگ قلم در يك استيل درجا (inline)

مثال کاربردی در مورد Style Sheet ها

مثالی در مورد نحوه تعريف استيلهای مختلف نمايش متون در جداول

تگ

تگ سبب تعريف محدوده و گستره ای مشخص مىشود. تمامی المان هائی كه درون تگهای و قرار خواهند

Learn HTML and CSS with w3schools | 2012

گرفت خواص آنرا به ارث خواهند برد، خواصی از قبیل استیل (Style) و یا سمت و جهت نمایش متون (dir). بر خلاف تگ های <p> و <div> مرورگرها هیچ سطر جدیدی به اول یا آخر المان اضافه نمی کنند.

مثال:

کد اجتممل	نمایش کد روبرو توسط مرورگر
<pre><p>Some text another text again text...</p></pre>	Some text another text again text...
id, class, title, style, dir, lang	شناسه های استاندارد المان span

مثالهایی از نحوه استفاده از المان span و Style sheet ها:

روش قدیمی	روش صحیح
<pre>Hello!</pre>	<pre>Hello!</pre>
<pre>Hello!</pre>	<pre>Hello!</pre>
<pre>Hello!</pre>	<pre>Hello!</pre>
<pre>Hello!</pre>	<pre>Hello!</pre>

تگ <div>

تگ <div> سبب تعریف محدوده و گستره ای مشخص میشود. تمامی المان هایی که درون تگهای <div> و </div> قرار خواهند گرفت خواص آنرا به ارث خواهند برد، خواصی از قبیل استیل (Style) و یا سمت و جهت نمایش متون (dir). بر خلاف تگ مرورگرها یک سطر جدید به اول و آخر المان <div> اضافه می کنند.

مثال:

کد اجتممل	نمایش کد روبرو توسط مرورگر
<pre><p>Some text <div style="color:#0000AF;">another text</div> again text...</p></pre>	Some text another text again text...
id, class, title, style, dir, lang	شناسه های استاندارد المان div

تعیین جهت نمایش متون

مثالی در مورد استفاده از تگ div و اعمال "جهت نمایش متون" به بخشی از صفحه اجتممل

Start Tag	Purpose	کاربرد
<div>	Defines a division/section in a document	تعریف بخش و ناحیه ای از صفحه اجتممل

 Defines a section in a document تعریف بخش و ناحیه ای از صفحه اچتمل

مرجع : HTML 4.01 Specification - فهرست عناصر - فهرست شناسه ها

استایل شیتها (Style Sheets)

در نسخه جدید(4/0) اچتمل امکان جداسازی کامل فرمت دهی از ساختار صفحات اچتمل فراهم شده است. در این نسخه توصیه میشود که تمامی اطلاعات مربوط به نمایش و ترازبندی به فایل‌های مجزا از صفحه اچتمل منتقل شوند. بخشی از صفحات اچتمل که مربوط به نمایش، ترازبندی و فرمت دهی است، استایل شیت یا "Style Sheet" نامیده شده و معمولا به صورت فایلی مجزا درون بخش <head> صفحات اچتمل معرفی میگردند. در این درس به صورت بسیار خلاصه با روشهای مختلف استفاده از استایل شیتها در صفحات اچتمل آشنا خواهید شد.

درس قبلی، مثالهای اچتمل

مثالها

استیلها در اچتمل

در این مثال نحوه تعریف استایل شیتها درون بخش <head> صفحات اچتمل شرح داده شده است. (Internal Style Sheet)

پیوندی بدون خط افقی زیری

در این مثال نحوه ایجاد پیوندی بدون "خط افقی زیری" شرح داده شده است. (Inline Styles)

اتصال به فایل‌های استایل شیت خارجی

در این مثال نحوه استفاده از تگ <link> جهت اتصال به فایل‌های استایل شیت خارجی شرح داده خواهد شد. (External Style Sheet)

چگونگی استفاده از استایل شیتها:

مروگرها از روی استیل‌های تعریف شده توسط استایل شیتها به فرمت دهی و نمایش اطلاعات درون صفحه اچتمل میپردازند. تعریف و استفاده از استایل ها در صفحات اچتمل به سه روش مختلف ممکن میباشد:

۱) استایل شیت‌های خارجی (external style sheet) :

در این روش تمامی استیلها و تعاریف نمایشی درون فایلی جداگانه قرار گرفته و بسیار مناسب حالت‌هایی است که قرار است که استایلی به بیش از یک صفحه اچتمل اعمال شود. مزیت این روش سهولت در تغییر ظاهر صفحات یک وب سایت میباشد و برای یک تغییر کلی در سایت کافی است که فایل استایل شیت کل سایت که معمولا یک فایل واحد است تغییر داده شود. (باید توجه داشت که در این حالت تمامی صفحات سایت باید به یک فایل استایل شیت واحد لینک شده باشند). استایل شیت‌های خارجی توسط تگ <link> که درون بخش head صفحات اچتمل قرار میگیرد، معرفی میشوند.

در مثال زیر با کمک شناسه href از تگ link نام و در حالت کلی url فایل در برگیرنده استایل‌های تعریف شده تعیین میگردد:

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

۲) استایل شیت‌های داخلی (Internal Style Sheet) :

کاربرد این روش در حالت‌هایی است که صفحه اچتمل نیازمند تعریف استایلی منحصر به فرد و مجزا بوده و استایل طراحی شده فقط بدرد آن صفحه خواهد خورد. در این حالت باید کدهای حاوی تعریف استایل درون تگ های `<style>` و `</style>` قرار گرفته و معمولا نتیجه درون بخش `head` صفحه اچتمل گذاشته میشود. مثال:

```
<head>
<style type="text/css">
body {background-color: red}
p {margin-left: 20px}
</style>
</head>
```

۳) استایل شیت‌های درجا (Inline Styles) :

کاربرد این روش در مواردی است که باید برای یک المان (یا گروهی) خاص استایلی اعمال شود. در این حالت باید از شناسه ای به نام `style` که تقریبا در تمامی تگها قابل اعمال است استفاده شود. مثال:

```
<p style="color: red; margin-left: 20px">
This is a paragraph
</p>
```

تگ های استایل

Start Tag Purpose

<code><style></code>	Defines a style in a document
<code><link></code>	Defines the relationship between two linked documents
<code></code>	Deprecated. Use styles instead
<code><basefont></code>	Deprecated. Use styles instead
<code><center></code>	Deprecated. Use styles instead

کاربرد

تعریف استیلها درون یک سند
تعیین و تعریف فایل استایل شیت خارجی
کنار گذاشته شده.از استایل شیتها استفاده شود.
کنار گذاشته شده.از استایل شیتها استفاده شود.
کنار گذاشته شده.از استایل شیتها استفاده شود.

مرجع : HTML 4.01 Specification - فهرست عناصر - فهرست شناسه ها

تغییرات در نسخه جدید اچتمل (4/0) :

هدف اولیه از طراحی زبان اچتمل ایجاد زبانی برای تعریف اطلاعات محض بدون در برداشتن جزئیات مربوط به ظاهر و نمایش بود ولی با ایجاد نسخه 3/2 زبان اچتمل و اضافه شدن تگهای فرمت دهی مانند `` و یا شناسه هائی مانند `color` و `Align` محتوا و نمایش در صفحات اچتمل ادغام گردیده و همین امر سبب بروز مشکلات عدیده ای برای طراحان و برنامه نویسان وب گردید. برای رفع این مشکل در نسخه 4/0 اچتمل توصیه بر جداسازی کامل این دو گروه از اطلاعات بوده و در نسخه های آینده اچتمل این جداسازی اجباری خواهد گردید و توصیه مدرسه وب نیز استفاده از استایل شیتها برای هر نوع تعریف فرمت و ترازبندی میباشد. برای مثال در کد اچتمل زیر که بر اساس نسخه 3/2 اچتمل میباشد، ساختار و محتوا (تگی از نوع پاراگراف با محتوای "This is a paragraph") با اطلاعات نمایش یعنی رنگ و ترازبندی (رنگ قرمز متن و ترازبندی در وسط صفحه) همزمان و در کنار هم آورده شده اند و این ادغام امکان تغییر و توسعه کد را بسیار محدود خواهد کرد:

```
<p align="center"><font color="red">This is a paragraph</font></p>
```

کد معادل مثال فوق با استفاده از استایل شیتها در این مثال شرح داده شده است. (هر چند که این مثال از تمامی امکانات استایل شیتها و ایجاد فایل کاملاً مستقل برای تعریف استایل استفاده نکرده است.)

استایل شیتها (Style Sheets)

در نسخه جدید (4/0) اچتمل امکان جداسازی کامل فرمت دهی از ساختار صفحات اچتمل فراهم شده است. در این نسخه توصیه میشود که تمامی اطلاعات مربوط به نمایش و ترازبندی به فایلهایی مجزا از صفحه اچتمل منتقل شوند. بخشی از صفحات اچتمل که مربوط به نمایش، ترازبندی و فرمت دهی است، استایل شیت یا "Style Sheet" نامید شده و معمولاً به صورت فایل مجزا درون بخش `<head>` صفحات اچتمل معرفی میگرددند. در این درس به صورت بسیار خلاصه با روشهای مختلف استفاده از استایل شیتها در صفحات اچتمل آشنا خواهید شد.

مثالها

استیلهما در اچتمل

در این مثال نحوه تعریف استایل شیتها درون بخش `<head>` صفحات اچتمل شرح داده شده است. (Internal Style Sheet)

پیوندی بدون خط افقی زیری

در این مثال نحوه ایجاد پیوندی بدون "خط افقی زیری" شرح داده شده است. (Inline Styles)

اتصال به فایلهای استایل شیت خارجی

در این مثال نحوه استفاده از تگ `<link>` جهت اتصال به فایلهای استایل شیت خارجی شرح داده خواهد شد. (External Style Sheet)

چگونگی استفاده از استایل شیتها:

مرورگرها از روی استیل‌های تعریف شده توسط استایل شیتها به فرمت دهی و نمایش اطلاعات درون صفحه اچتمل میپردازند. تعریف و استفاده از استایل ها در صفحات اچتمل به سه روش مختلف ممکن میباشد:

۱) استایل شیت‌های خارجی (external style sheet) :

در این روش تمامی استیلها و تعاریف نمایشی درون فایل جداگانه قرار گرفته و بسیار مناسب حالت‌هایی است که قرار است که استایلی به بیش از یک صفحه اچتمل اعمال شود. مزیت این روش سهولت در تغییر ظاهر صفحات یک وب سایت میباشد و برای یک تغییر کلی در سایت کافی است که فایل استایل شیت کل سایت که معمولا یک فایل واحد است تغییر داده شود. (باید توجه داشت که در این حالت تمامی صفحات سایت باید به یک فایل استایل شیت واحد لینک شده باشند.) استایل شیت‌های خارجی توسط تگ `<link>` که درون بخش `head` صفحات اچتمل قرار میگیرد، معرفی میشوند. در مثال زیر با کمک شناسه `href` از تگ `link` نام و در حالت کلی `url` فایل در برگیرنده استایل‌های تعریف شده تعیین میگردد:

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

۲) استایل شیت‌های داخلی (Internal Style Sheet) :

کاربرد این روش در حالت‌هایی است که صفحه اچتمل نیازمند تعریف استایلی منحصر به فرد و مجزا بوده و استایل طراحی شده فقط بدر آن صفحه خواهد خورد. در این حالت باید کدهای حاوی تعریف استایل درون تگ های `<style>` و `</style>` قرار گرفته و معمولا نتیجه درون بخش `head` صفحه اچتمل گذاشته میشود. مثال:

```
<head>
<style type="text/css">
body {background-color: red}
p {margin-left: 20px}
</style>
</head>
```

۳) استایل شیت‌های درجا (Inline Styles) :

کاربرد این روش در مواردی است که باید برای یک المان (یا گروهی) خاص استایلی اعمال شود. در این حالت باید از شناسه ای به نام `style` که تقریبا در تمامی تگها قابل اعمال است استفاده شود. مثال:

```
<p style="color: red; margin-left: 20px">
This is a paragraph
</p>
```

تگ های استایل

Start Tag Purpose

کاربرد

<code><style></code>	Defines a style in a document	تعریف استیلهای درون یک سند
<code><link></code>	Defines the relationship between two linked documents	تعیین و تعریف فایل استایل شیت خارجی
<code></code>	Deprecated. Use styles instead	کنار گذاشته شده. از استایل شیتها استفاده شود.
<code><basefont></code>	Deprecated. Use styles instead	کنار گذاشته شده. از استایل شیتها استفاده شود.
<code><center></code>	Deprecated. Use styles instead	کنار گذاشته شده. از استایل شیتها استفاده شود.

مرجع : HTML 4.01 Specification - فهرست عناصر - فهرست شناسه ها

تغییرات در نسخه جدید اچتمل (4/0) :

هدف اولیه از طراحی زبان اچتمل ایجاد زبانی برای تعریف اطلاعات محض بدون در برداشتن جزئیات مربوط به ظاهر و نمایش بود ولی با ایجاد نسخه 3/2 زبان اچتمل و اضافه شدن تگهای فرمت دهی مانند `` و یا شناسه هائی مانند `color` و `Align` محتوا و نمایش در صفحات اچتمل ادغام گردیده و همین امر سبب بروز مشکلات عدیده ای برای طراحان و برنامه نویسان وب گردید. برای رفع این مشکل در نسخه 4/0 اچتمل توصیه بر جداسازی کامل این دو گروه از اطلاعات بوده و در نسخه های آینده اچتمل این جداسازی اجباری خواهد گردید و توصیه مدرسه وب نیز استفاده از استایل شیتها برای هر نوع تعریف فرمت و ترازبندی میباشد. برای مثال در کد اچتمل زیر که بر اساس نسخه 3/2 اچتمل میباشد، ساختار و محتوا (تگی از نوع پاراگراف با محتوای "This is a paragraph") با اطلاعات نمایش یعنی رنگ و ترازبندی (رنگ قرمز متن و ترازبندی در وسط صفحه) همزمان و در کنار هم آورده شده اند و این ادغام امکان تغییر و توسعه کد را بسیار محدود خواهد کرد:

```
<p align="center"><font color="red">This is a paragraph</font></p>
```

کد معادل مثال فوق با استفاده از استایل شیتها در این مثال شرح داده شده است. (هر چند که این مثال از تمامی امکانات استایل شیتها و ایجاد فایل کاملاً مستقل برای تعریف استایل استفاده نکرده است.)

تگهای اصلی (Basic Tags)	
<code><html></html></code>	ایجاد سندی اچتمل
<code><head></head></code>	ایجاد اطلاعات شناسنامه ای سند مانند عنوان و ...
<code><body></body></code>	ایجاد بدنه و قسمت قابل مشاهده سند اچتمل
تگهای ناحیه Header	
<code><title></title></code>	تعیین عنوان سند

شناسه های مهم تگ body	
<body bgcolor=?>	تعیین رنگ زمینه بر اساس نام رنگ یا معادل هگز
<body text=?>	تعیین رنگ نوشته بر اساس نام رنگ یا معادل هگز
<body link=?>	تعیین رنگ پیوندها بر اساس نام رنگ یا معادل هگز
<body vlink=?>	تعیین رنگ پیوندهای بازدیدشده بر اساس نام رنگ
<body alink=?>	تعیین رنگ پیوند فعال بر اساس نام رنگ یا معادل هگز
تگهای متن و نوشته	
<pre></pre>	نمایش متن های از قبل فرمت شده (preformatted)
<h1></h1>	ایجاد بزرگترین سر تیتر
<h6></h6>	ایجاد کوچکترین سر تیتر
	ایجاد متن توپر
<i></i>	ایجاد متن مورب و ایتالیک
<tt></tt>	ایجاد متن از نوع تله تایپی
<cite></cite>	ایجاد citation معمولاً ایتالیک
	نمایش تاکید شده متن (توپر یا مورب)
	نمایش تاکید شده متن (توپر یا مورب)
	تعیین اندازه قلم از ۱ تا ۷
	تعیین رنگ قلم بر اساس نام رنگ یا مقدار هگز آن
پیوندها	
	ایجاد پیوند
	ایجاد پیوندی از نوع mailto
	ایجاد پیوندی نامگذاری شده در یک سند
	ارجاع به یک پیوند نامگذاری شده
فرمت دهی	
<p></p>	ایجاد پاراگرافی جدید
<p align=?>	ترازبندی پاراگراف در سمت left, right, or center
 	رفتن سر سطر جدید
<blockquote> </blockquote>	عنوان سازی متن
<dl></dl>	ایجاد فهرستهای تعریفی
<dt>	ایجاد عنصری از یک فهرست تعریفی
<dd>	ایجاد شرح و توصیف یک عنصر فهرست تعریفی

<code></code>	ایجاد فهرستهای مرتب
<code></code>	ایجاد یک قلم اطلاعاتی از یک فهرست
<code></code>	ایجاد فهرستی نامرتب
<code><div align=?></code>	تگی برای فرمت دهی بلوکی بزرگ از کدهای اچتمل
	المانهای گرافیکی
<code></code>	ایجاد یک تصویر
<code></code>	تراز بندی تصویر (left, right, center; bottom, top,) (middle)
<code></code>	تعیین ضخامت مرز یک تصویر
<code><hr></code>	درج خطی افقی
<code><hr size=?></code>	تعیین ارتفاع خط افقی
<code><hr width=?></code>	تعیین عرض خط بر اساس مقدار واقعی یا درصدی
<code><hr noshade></code>	ایجاد خطی افقی بدون سایه
	جداول
<code><table></table></code>	ایجاد جدول
<code><tr></tr></code>	ایجاد ردیف در جدول
<code><td></td></code>	ایجاد خانه های جدول
<code><th></th></code>	ایجاد عنوان ستونهای یک جدول
	شناسه های جدول
<code><table border=#></code>	تعیین مقدار ضخامت مرزهای یک جدول
<code><table cellspacing=#></code>	تعیین فاصله بین خانه های جدول
<code><table cellpadding=#></code>	تعیین فاصله بین محتوای یک خانه و مرزهای آن
<code><table width=# or %></code>	تعیین عرض جدول بر اساس پیکسل و یا درصدی
<code><tr align=?> or <td align=?></code>	تعیین ترازبندی خانه های جدول (left, center, right)
<code><tr valign=?> or <td valign=?></code>	ترازبندی عمودی خانه های جدول (top, middle, bottom)
<code><td colspan=#></code>	تعیین تعداد ستون هائی که باید یکی شوند.
<code><td rowspan=#></code>	تعیین تعداد سطر هائی که باید یکی شوند.
<code><td nowrap></code>	باعث جلوگیری از شکسته شدن متن در خانه جدول میشود.
	فریمها
<code><frameset></frameset></code>	تعریف و ایجاد مجموعه ای از فریمها
<code><frameset rows="value,value"></code>	شناسه ای برای تعریف ردیفهای یک مجموعه فریم بر اساس اندازه

<code><frameset cols="value,value"></code>	واقعی بر حسب پیکسل و یا ابعاد درصدی شناسه ای برای تعریف ستونها یک مجموعه فریم بر اساس اندازه
<code><frame></code>	واقعی بر حسب پیکسل و یا ابعاد درصدی تعریف فریمی از مجموعه فریمها
<code><noframes></noframes></code>	تعیین متنی که باید در صورت عدم پشتیبانی مرورگر از فریمها باید نمایش داده شود.
<code><frame src="URL"></code>	شناسه های فریمها تعیین سند اچتمل داخل فریم
<code><frame name="name"></code>	تعیین نام برای یک فریم
<code><frame marginwidth=#></code>	تعیین عرض حاشیه راست و چپ فریم
<code><frame marginheight=#></code>	تعریف طول حاشیه بالا و پایین فریم بر اساس پیکسل
<code><frame scrolling=VALUE></code>	تعیین اینکه فریمی از Scroll bar استفاده کند یا نه. مقادیر ممکن: yes, no و auto (حالت پیش فرض)
<code><frame noresize></code>	سبب جلوگیری از تغییر اندازه یک فریم میشود. فرمها
<code><form></form></code>	ایجاد یک فرم
<code><select multiple name="NAME" size=?></select></code>	ایجاد یک منوی کرکره ای و تعیین تعداد اقلامی که قبل از scrolling نمایش داده خواهند شد.
<code><option></code>	تعیین هر یک از اقلام منو کرکره ای
<code><select name="NAME"></select></code>	ایجاد منوی کرکره ای
<code><option></code>	تعیین هر یک از اقلام منو کرکره ای
<code><textarea name="NAME" cols=40 rows=8></textarea></code>	ایجاد المانی برای ورود متن در بیش از یک خط با تعیین تعداد سطر و ستون ناحیه
<code><input type="checkbox" name="NAME"></code>	ایجاد چک باکسی با تعیین متن و عنوان آن
<code><input type="radio" name="NAME" value="x"></code>	ایجاد رادیو باتن با تعیین متن و عنوان آن
<code><input type="text" name="foo" size=20></code>	ایجاد المانی برای ورود متن با حداکثر یک سطر به همراه تعیین ابعاد بر اساس نویسه
<code><input type="submit" value="NAME"></code>	ایجاد دکمه ای از نوع ارسال و Submit
<code><input type="image" border=0 name="NAME" src="name.gif"></code>	ایجاد دکمه ای از نوع ارسال با کمک یک تصویر
<code><input type="reset"></code>	ایجاد دکمه خلاصی !! یا Reset

مراجع و رفرنسها

فرهنگ لغت واژه های فنی در وب

Anchor	پیوند، لینک، (لنگر کشتی)
Attribute	شناسه، نشان، صفت
Browser, user agent	مرورگر، شبکه نورد، بروزر
Caption	عنوان ، سرلوحه
Character	نویسه، کاراکتر، دخشه
Client	سرویس گیرنده، مشتری ، ارباب رجوع
DHTML	اچتمل دینامیک، FAQ
Division, Div	بخش، قسمت، دسته بندی
Download	گرفتن اطلاعات از وب سرور
Document	سند، متنی اچتمل
Edit	ویرایش
Element	المان، عنصر
Explorer	جستجوگر، مکتشف
Extention	انشعاب فایل
FAQ	بیشترین سؤالات پرسیده شده در مورد یک موضوع
Font	قلم، خانواده حروف
Format	قالب، فرمت
Frame	قاب، چهارچوب
Frameset	مجموعه ای از فریمها
FTP	استانداردی برای مبادله فایل (Protocol Transfer File)
href	آدرس متن مختلط (reference hyerlink)
IP	Protocol Internet
JavaScript	جاوا اسکریپت، زبانی برای اسکریپت نویسی
HTTP	استاندارد تبادل ابر متن ها (Protocol Transfer Hypertext)
Hyper text	ابر متن، متن مختلط
List	فهرست

Link	پیوند، لینک
Online	آنلاین، سرخط
Padding	لایه گذاری، لفافه
Page, Web page	صفحه، متن
Publish	انتشار، بردن اطلاعات روی وب سرور
Protocol	استاندارد ارتباطی، فهرست پروتوکلهای مهم در اینترنت
Robot	نرم افزاری خودکار در وب
Site, Web site	پایگاه، سایت
Script	اسکرپت، دستخط، متن نمایشنامه
Spacing	فاصله گذاری، فاصله
Span	محدوده، گستره، وجب
Target	هدف و مقصد
Table	جدول، آرایه
Tag	تگ، برچسب
TCP/IP	نوعی استاندارد ارتباطی (Transmission Control Protocol/Internet Protocol)
Unicode	استانداردی برای محیط های چند زبانه
Upload	بردن اطلاعات روی وب سرور، معمولا با کمک نرم افزارهای FTP انجام میپذیرد.
URL	آدرس اینترنتی، خلاصه شده Locator Resource Uniform
User, Surfer	کاربر، وبگرد، بازدیدکننده
Weblog	وب نویسی، تارنگاری
Webmaster	مدیریت سایت
Web Server	سرویس دهنده وب، نرم افزاری که وظیفه اصلیش دادن سرویس به درخواستهای دریافت شده از وب میباشد.
Web, WWW	وب، تار عنکبوت جهانی، خلاصه شده Web Wide World

منبع اصلی : w3school.com
برگرفته شده از سایت khaterat.com



در کانال تلگرام کارنیل هر روز انگیزه خود را شارژ کنید 😊

<https://telegram.me/karnil>

