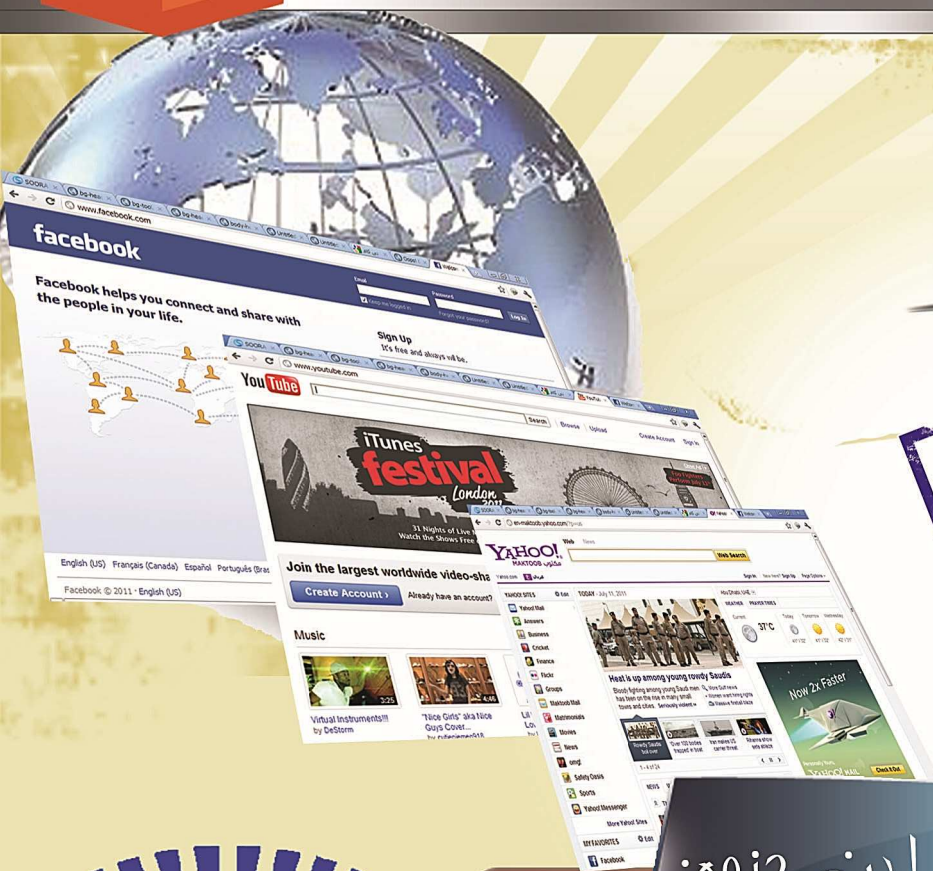


HTML

5

مرجع آموزش HTML و XHTML

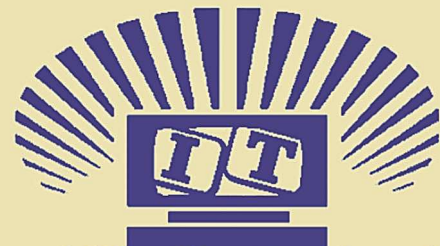


همراه با پوست
HTML 5

در این جزوه:

بررسی انواع نسخه ها و تفاوت های HTML و XHTML
آشنایی با مفاهیم Tag و Attribute
آشنایی با انواع تگ ها ، صفت ها و کاربردهایشان
آشنایی با به کارگیری لیست ها ، جداول و فرم ها در صفحات وب
آشنایی با به کارگیری عناصر چند رسانه ای
آشنایی با نحوه ایجاد صفحات چند گانه (Frameset)
ارائه انوان مثال ها و نمونه کد ها

نویسنده: احمد بادپی



انجمن علمی مهندسی فناوری اطلاعات
دانشگاه پیام نور مرکز آران و بیدگل

پیشگفتار:

به مرجع آموزش HTML و XHTML خوش آمدید.

در طول حیات نسبتاً کوتاه وب ، تکنولوژی های مختلفی به منظور ایجاد صفحات اینترنتی معرفی شده اند. اما در این میان نمی توان از نقش مهم و اساسی HTML به عنوان شالوده و آغازگر وب که در عین سادگی در یادگیری از اهمیت فراوانی در خلق صفحات اینترنتی برخوردار است چشم پوشی کرد. اگر از من بپرسید می گویم تسلط بر تمامی مفاهیم و جنبه های این زبان نشانه گذاری برای یک طراح وب از نان شب هم واجب تر بوده! و برای خلق صفحات اینترنتی زیبا و منطبق بر اصول از ملزومات ورود به عرصه طراحی وب است.

جزوه پیش رو که در راستای برگزاری دوره های آموزشی طراحی وب به همت انجمن علمی مهندسی فناوری اطلاعات دانشگاه پیام نور مرکز آران و بیدگل تهیه شده است شما را با بسیاری از ویژگی های این زبان پرترفدار و دوست داشتنی آشنا خواهد کرد.

این جزوه از روی برترین و محبوب ترین مراجع آموزش HTML حال حاضر دنیا همچون کتاب *The Ultimate HTML Reference* نوشته Lan Loyd ، *Beginning HTML , XHTML , CSS and Javascript* نوشته jon Duckett ، سایت پر آوازه ی www.w3schools.com و البته تجربیات شخصی ترجمه و تالیف شده است . همچنین لازم به ذکر است به منظور پوشش آخرین نسخه ارائه شده از این زبان یعنی HTML 5 از کتاب *AbookApart HTML5 For Web Designers* نوشته jeremy keith کمک گرفته شده است.

ذکر این نکته نیز ضروری است که ترجمه، تخصص اینجانب نبوده و هدف از تهیه جزوه ای که هم اکنون در دستان شماست هم چیزی جز جامه عمل پوشاندن به حدیث شریف «رَكَاةُ الْعِلْمِ كَثْرَةٌ» نبوده است. از این رو این جزوه نیز خالی از اشکال و اشتباه نیست. لذا از تمامی خوانندگان تقاضا می شود به منظور گزارش اشکال و بیان نظرات، انتقادات و پیشنهادات خود با ایمیل اینجانب به آدرس ahmadbadpey@gmail.com مکاتبه فرمایند.

در پایان از تمامی دوستانم علی الخصوص مجید بصیرتی به خاطر ویرایش و صفحه آرایی ، رامین محققى به خاطر طراحی طرح روی جلد و تمامی اعضای محترم انجمن علمی مهندسی فناوری اطلاعات دانشگاه پیام نور مرکز آران و بیدگل و تمامی دوستانی که با اعتماد به بنده و حضور گرم خود در دوره های آموزشی برگزار شده موجبات دلگرمی و انگیزه برای تهیه این جزوه را فراهم آوردند صادقانه و صمیمانه تشکر می کنم.

به امید بهروزی

احمد بادپی - تیرماه ۱۳۹۰

فصل اول : آشنایی با مفاهیم و اصطلاحات

۱

فصل دوم : تاریخچه مختصری از HTML

۵

۵..... HTML چیست ؟

۶..... تاریخچه HTML

۶..... ویژگی های برجسته HTML 4.0

۷..... XHTML در مقابل HTML

۷..... HTML 5 از راه رسید!

فصل سوم : اجزای سازنده صفحات وب

۹

۱۰..... ساختار Tag ها

۱۱..... ساختار اصلی یک صفحه وب

۱۱..... • عنصر <head>

۱۱..... • عنصر <body>

۱۳..... ایجاد و ذخیره اولین صفحه وب

۱۴..... صفت ها در HTML

۱۶..... مشاهده کدهای منبع صفحات

فصل چهارم : آشنایی با تگ های قالب بندی متن

۱۷

۱۸..... آشنایی با انواع تگ های قالب بندی متن

۱۸..... • تگ

۱۸..... • تگ <i>

۱۸..... • تگ <u>

۱۸..... • تگ های <s> و <strike>

۱۸..... • تگ <sup>

۱۸..... • تگ <sub>

۱۹..... • تگ <big>

۱۹..... • تگ <small>

۱۹..... • تگ های و <ins>

۲۰..... • تگ های <abbr> و <acronym>

۲۰..... • ایجاد تیتراها با استفاده از تگ های <h1>

۲۲..... • ایجاد پاراگراف ها با استفاده از تگ <p>

۲۲..... • تگ
 برای ایجاد شکست خط

۲۲..... • تگ <nobr>

۲۳..... • تگ های <pre> و <code>

- تگ <hr /> ۲۳
- تگ <blink> ۲۴
- تگ <center> ۲۴
- تگ ۲۴
- تگ <basefont> ۲۵
- تگ <marquee> ۲۵
- عناصر بلاکی و درون خطی ۲۶

فصل پنجم : کار با لینک ها ۲۹

- تگ <a> ۳۰
- اجزای تشکیل دهنده آدرس ها ۳۱
- آدرس های مطلق و نسبی ۳۱
- تگ <base> ۳۲
- لینک های ایمیلی ۳۲
- لینک های لنگری ۳۴
- لینک های دانلود ۳۷

فصل ششم : کار با تصاویر ۳۹

- تگ ۴۰
- تصاویر به عنوان لینک ۴۱
- درج توضیحات در HTML ۴۱

فصل هفتم : کار با لیست ها ۴۳

- دسته بندی لیست ها ۴۴
- تگ برای ایجاد لیست های مرتب ۴۵
- تگ برای ایجاد لیست های نامرتب ۴۵
- تگ <dl> برای ایجاد لیست های تعریفی ۴۶
- تگ ۴۷
- لیست های تو در تو ۴۸

فصل هشتم : کار با جداول ۵۱

- ساختار اساسی جدول ۵۲
- تگ <table> ۵۲
- تگ <tr> ۵۴
- تگ های <td> و <th> ۵۵
- تگ <caption> ۵۸
- گروه بندی ردیفی و ستونی جداول ۵۸

گروه بندی های ستونی ۵۹

فصل نهم : کار با فرم ها ۶۳

ساختار اساسی فرم ۶۴

• تگ <form> ۶۵

• تگ <label> ۶۷

• تگ <input> ۶۸

صفت های جدید برای تگ های input ۷۴

ایجاد combobox و listbox ها ۷۶

• تگ <select> ۷۶

• تگ <option> ۷۶

• تگ <optgroup> ۷۷

ایجاد لیست های داده ۷۸

ایجاد کادر های متنی چند خطی ۷۹

• تگ <textarea> ۷۹

ایجاد انواع دکمه ها ۷۹

• تگ <button> ۷۹

سازمان دهی عناصر فرم ۸۰

• تگ <fieldset> ۸۰

• تگ <legend> ۸۰

ایجاد نوار های پیشرفت ۸۱

فصل دهم : عناصر چند رسانه ای در صفحات وب ۸۳

تعریف عناصر چند رسانه ای ۸۴

بررسی انواع عناصر چند رسانه ای ۸۴

• تگ Object ۸۶

• تگ param ۸۷

راه های دیگر برای استفاده از عناصر چند رسانه ای ۸۸

• تگ <embed> ۸۸

• تگ <bgsound> ۸۹

استفاده از ویژگی های HTML 5 برای نمایش صدا و ویدئو ۸۹

• تگ <video> ۹۰

• تگ <audio> ۹۱

فصل یازدهم : ایجاد صفحات چند گانه در HTML ۹۳

ساختار صفحات چند گانه ۹۴

• تگ <FRAMESET> ۹۴

- تگ <frame /> ۹۵
- تگ <noframes> ۹۶
- تگ <iframe> ۹۶

فصل دوازدهم : قرار دادن سایت بر روی وب

- تگ <meta /> ۱۰۰
- گرفتن نام دامنه ۱۰۳
- قرار دادن سایت بر روی سرور از طریق FTP ۱۰۵
- کار با برنامه CuteFtp ۱۰۶

ضمیمه

- تفاوت های HTML و XHTML ۱۱۱
- MIME TYPE چیست ؟ ۱۱۲
- کد های زبانی ۱۱۲
- کد های کاراکتری ۱۱۳

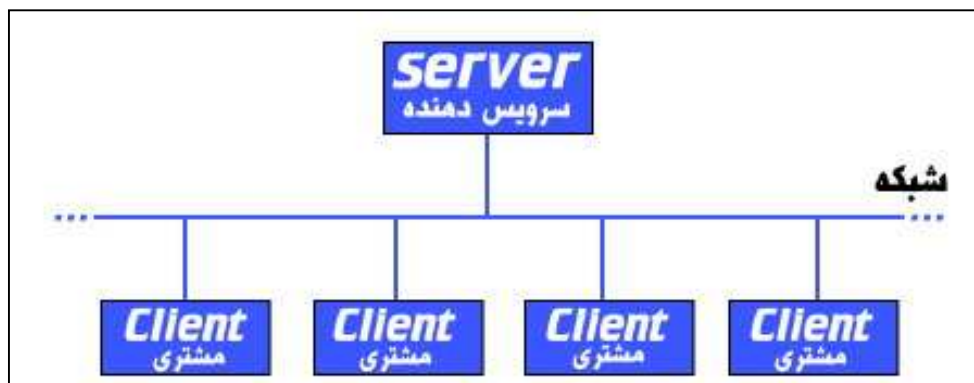
فصل اول

آشنایی با مفاهیم و اصطلاحات

در این فصل با مفاهیم و اصطلاحات رایج موجود در طراحی وب و اینترنت آشنا خواهیم شد. این مفاهیم در سراسر این کتاب بارها و بارها تکرار شده اند و یادگیری آن ها همواره از ملزومات علم طراحی وب بوده و به ما در فراگیری آسان تر مطالب کمک خواهند کرد.

اینترنت

اگر بخواهیم اینترنت را به زبان ساده بیان کنیم می‌توان آن را شبکه‌ای از کامپیوترها که یکسری اطلاعات را بین یکدیگر منتقل می‌کنند در نظر گرفت. به مدل ساده زیر نگاه کنید:



شکل ۱-۱

منابع اطلاعاتی^۱

همان اطلاعاتی هستند که بین کامپیوترهای موجود در اینترنت منتقل می‌شوند. این اطلاعات می‌توانند شامل هرگونه فایل با فرمت‌های گوناگون باشند. مانند عکس‌ها، فیلم‌ها، صداها، متن‌ها و... کامپیوترهای داخل شبکه اینترنت را به دو دسته تقسیم می‌کنند:

سرویس دهنده^۲

کامپیوترهایی که منابع اطلاعاتی را در اختیار دارند و وظیفه دارند که آن‌ها در اختیار سایر کامپیوترها قرار دهند سرویس دهنده نامیده می‌شوند. در واقع این کامپیوترها وظیفه سرویس دهی به سایر کامپیوترها را بر عهده دارند.

سرویس گیرنده^۳

به کامپیوترهایی که منابع اطلاعاتی را از سرویس دهنده‌ها دریافت می‌کنند سرویس گیرنده (مشتری) می‌گویند. اصطلاحاً به این مکانیزم و ساختار در اینترنت که یک کامپیوتر نقش سرویس دهنده و کامپیوترهای دیگر نقش سرویس گیرنده را بازی می‌کنند معماری **مشتری-سرویس دهنده**^۴ گفته می‌شود.

سایت

مجموعه‌ای از چندین فایل با انواع گوناگون همچون فایل‌های متنی، تصویری، صوتی و... که در واقع همان منابع اطلاعاتی روی اینترنت هستند.

¹ Information Resource

² server

³ Client

⁴ server-client

هاست^۱

به مقدار فضایی از حافظه سرویس دهنده ها که برای ذخیره منابع اطلاعاتی مورد استفاده قرار می گیرد گفته می شود. به سروری که محتویات یک سایت را در خود نگه داری می کند وب سرور آن سایت می گویند.

Request

به درخواستی که از طرف سرویس گیرنده ها برای دریافت یک فایل به سرویس دهنده ای خاص ارسال می شود گفته می شود.

Response

به پاسخی که یک سرویس دهنده در جواب درخواست یک سرویس گیرنده می دهد گفته می شود.

مرورگر^۲

به نرم افزاری که وظیفه ارسال درخواست ها به سرویس دهنده و دریافت پاسخ ها از آن را بر عهده دارد مرورگر گفته می شود. از رایج ترین مرورگرها می توان از Internet Explorer، Mozilla Firefox، Opera، Safari و Google Chrome نام برد. به مرورگرها عامل کاربر^۳ نیز می گویند.

Download

به عمل انتقال یک فایل از server به client گفته می شود.

Upload

به عکس عمل download یعنی انتقال فایل از client به server گفته می شود.

W3C^۴

به سازمان بین المللی استانداردسازی وب گفته می شود. این سازمان که مدیریت آن را بر عهده مخترع وب یعنی تیم برنرزلی است وظیفه ارائه استاندارد ها و تکنولوژی های مرتبط با توسعه وب همچون HTML و CSS را بر عهده دارد.

¹ Host

² Web browser

³ UserAgent

⁴ World Wide Web Consortium

فصل دوم

تاریخ مختصری از HTML

HTML چیست ؟

برای ارسال منابع اطلاعاتی در پهنه وسیعی از مخاطبان نیاز به یک زبان واحد جهانی است که برای تمام کامپیوترها قابل فهم باشد. این زبان HTML است.

HTML به نویسندگانش امکانات زیر را می‌دهد:

- به اشتراک گذاری منابع اطلاعاتی به صورت آنلاین و از طریق متن‌ها، جدول‌ها، لیست‌ها، عکس‌ها و...
- دریافت آن‌ها از طریق لینک‌ها و کلیک کردن بر روی یک دکمه
- طراحی فرم‌ها به منظور تبادل اطلاعات با کامپیوترهای راه دور و به جهت جستجو، ذخیره، و مرتب‌سازی آنها و...
- قرار دادن کلیپ‌های ویدئویی، صوتی و دیگر برنامه‌های کاربردی در داخل اسنادشان

تاریخچه HTML

HTML توسط Tim Berners-Lee در خلال دهه 90 میلادی همراه با گسترش وب، شکوفا شد. این زبان توسط مرورگر Mosaic معروفیت خاصی پیدا کرد. در آن زمان HTML در چند مدل منتشر می شد که آن بستگی داشت به سازنده فایل و انجمن‌هایی که در زمینه وب فعالیت داشتند.

در نوامبر 1995 نسخه HTML 2.0 گسترش یافت و بلافاصله در همان سال HTML 3.0 منتشر شد، ولی استقبالی از آن نشد. در سال 1996 انجمن W3C شروع به فعالیت بر روی نسخه این زبان کرد که حاصل کار آنها در 14 ژانویه 1997 انتشار HTML 3.02 بود. این نسخه توانست رضایت اکثریت را جلب کند چون هماهنگی بیشتری با مرورگرهای مختلف در سیستم‌های عامل متفاوت داشت. در تمام نسخه های این زبان، سعی بر این شده بود تا نظر کسانی که در زمینه وب سرمایه گذاری کرده بودند جلب شود و برنامه های تولید شده برای وب بتوانند مدت طولانی تری قابل استفاده باشند. به همین منظور HTML برای اهداف گسترده تری، در وب توسعه یافت تا در کلیه سیستم‌های اطلاع رسانی و الکترونیکی کوچک و بزرگ با بکار بردن گرافیک و رنگها، قابلیت بهره برداری بیشتری داشته باشد.

در 18 دسامبر 1997 نسخه HTML 4.0 در وب منتشر شد و در همین بین شرکتهای تولید کننده مرورگر وب یک سری مشخصات منحصر به خود را به این نسخه اضافه کردند که قابل اجرا در مرورگرهای دیگر نبود. بعضی از این تغییرات در W3C مورد تایید قرار گرفت اما بعضی دیگر نه. با تغییرات HTML مرورگرها مجبور به تغییر شدند تا با تحولات جدید سازگار شوند.

در تاریخ 24 آوریل 1998 در این نسخه تجدید نظر شد و حاصل آن پیدایش HTML 4.01 بود که با کمی تغییر و رفع یکسری مشکلات، در W3C برسمیت شناخته شد و این انجمن استفاده از آنرا به توسعه دهندگان و طراحان وب، توصیه کرد.

ویژگی های برجسته HTML 4.0

بطور کل مجموعه HTML 4 به دلیل ویژگی هایی که نسبت به نسخ قبلی داشت توانست محبوبیت و مقبولیت خاصی در بین طراحان پیدا کند که در ادامه به آن ها اشاره خواهیم کرد:

پشتیبانی از انواع زبان ها و کاراکترها

یکی از مهمترین ویژگی های مجموعه HTML 4 پشتیبانی آن از انواع زبان های جهان بود. تا قبل از آن با HTML فقط امکان طراحی صفحات به زبان های چپ به راست مثل انگلیسی و فرانسوی و... بود. اما این نسخه امکان استفاده از زبان های راست به چپ و کاراکترهای منحصر به فرد آن ها همچون فارسی و عربی را نیز فراهم کرد.

استفاده از برگه های سبک آبشاری¹

یکی دیگر از ویژگی ها پشتیبانی خوب آن از stylesheet ها برای کنترل چگونگی نمایش صفحات توسط کاربران و طراحان بود. تا قبل از آن، طراحان کنترل بسیار کمی بر نحوه نمایش صفحات خود داشتند.

¹ Cascading Style Sheets

گنجاندن زبان های اسکریپتی همچون javascript، vbscript و...

یکی دیگر از ویژگی ها امکان گنجاندن زبان های اسکریپتی در داخل صفحات HTML برای خلق صفحات پویا^۱ بود. به وسیله این زبان ها می توان بر نحوه رفتار عناصر موجود در یک صفحه وب کنترل داشت و تعیین کرد که عناصر در مقابل کنش های کاربر صفحه چه واکنشی از خود نشان دهند.

جداول^۲

یکی از عناصری که امکان استفاده از آن در HTML 4.0 فراهم شد امکان به کارگیری جداول برای نظم دهی و ارائه منظم و قالب بندی شده اطلاعات بود.

XHTML در مقابل HTML

HTML 4 زبانی بسیار قوی برای طراحان و سازندگان صفحات وب است اما در آن توجهی به دستگاههای اطلاع رسانی و الکترونیکی کوچک با قدرت و حافظه کمتر نشده است. به همین منظور W3C در ۲۶ ژانویه ۲۰۰۰ اقدام به معرفی XHTML 1.0 کرد و در ۱۹ دسامبر همان سال آن را به رسمیت شناخت و تاکید بر استفاده از این نسخه کرد تا با به کارگیری ویژگی هایی که دارد دامنه استفاده از زبان محبوب HTML را گسترش دهد و مقدمات معرفی XML را فراهم کند.

XHTML از نظر ساختاری با نسخ قبلی تفاوتی نکرده است و تنها دارای یکسری اصول و قوانین خاص است تا کمبودهای HTML را برطرف کند. XHTML مخفف عبارت eXtensible Hyper Text Markup Language و به معنی HTML قابل توسعه (توسعه پذیر) است.

XHTML فقط شامل قوانین و اصولی است که طراحان وب باید در هنگام طراحی هایشان از آن ها پیروی کنند. به عنوان مثال یکی از این قوانین این است که در XHTML، دستورات می بایست حتما به صورت کوچک نوشته شوند (درواقع دستورات case sensitive بوده و نسبت به حروف حساس هستند) در حالی که خود HTML چنین محدودیتی را ندارد. برای اطلاع بیشتر در مورد تفاوت های نسخه XHTML با نسخ قبلی می توانید به بخش «تفاوت های HTML 4.0 و XHTML» که در ضمیمه این جزوه آمده است مراجعه نمایید.

HTML 5 از راه رسید!

HTML5 نسخه شماره ۵ زبان اصلی World Wide Web یعنی HTML است. این نسخه در تاریخ ۲۲ ژانویه ۲۰۰۸ بوسیله کنسرسیوم وب جهانی (W3C) منتشر شد. ایده ابتدایی این نسخه از HTML در سال ۲۰۰۴ در WHATWG کلید خورد. این گروه کاری شامل شرکت های بزرگی مانند AOL، Apple، Google، IBM، Microsoft، Mozilla، Nokia، Opera و ... است.

HTML 5 به عنوان جهت دهنده جدیدی برای توسعه وب معرفی شده است. مشخصه های جدید HTML5 بیشتر بر روی Web Application ها و معرفی صدا و تصویر بعنوان رسانه سطح اول در وب متمرکز شده است که با استاندارد سازی این قابلیت ها توسط W3C بالتبع فراگیری و کاربرد آنها هموار تر خواهد شد.

¹ Dynamic

² Tables

از دیگر قابلیت های پیش بینی شده در HTML 5، ارایه مشخصه هایی برای نمایش دادن عناصر آشنای صفحات مانند سرصفحه^۱، راهبری^۲ و پاورقی^۳ و ... می باشد.

مدیریت ذخیره اطلاعات در سمت کاربر نیز یکی دیگر از نکات برجسته HTML 5 است. این نسخه تگ ها و صفت های جدیدی را نیز معرفی کرده است که برخی از مهمترین آنها که در بسیاری از مرورگر ها پشتیبانی می شود را بررسی خواهیم کرد.

¹ Header

² Navigation

³ Footer

فصل سوم

اجزای سازنده صفحات وب

همانند بسیاری از زبان های برنامه سازی رایج ، HTML نیز به نوبه خود از اجزا و قواعد خاص خود در ایجاد صفحات وب تشکیل شده است. در این فصل ابتدا به سراغ بخش های تشکیل دهنده عبارت HTML خواهیم رفت و در ادامه با ساختار دستورات اساسی و مفاهیم این زبان همچون تگ ، صفت و مقادیر و سپس با ساختار اصلی صفحات اینترنتی و چند عنصر ابتدایی این زبان و نحوه ایجاد اولین صفحه اینترنتی آشنا خواهیم شد.

HTML زبان خلق صفحات وب است. بوسیله آن می توان صفحات وبی ساخت که در آن ها از عکس ها، فیلم ها، موزیک ها، پاراگراف ها، جداول و... استفاده کرد.
HTML مخفف عبارت زیر می باشد:

Hyper Text Markup Language

که در ادامه به توضیح هر یک از عبارت های به کار رفته در آن خواهیم پرداخت:
Language: این عبارت نشان دهنده این است که HTML یکی از زبان های برنامه نویسی است. (البته در ادامه خواهیم دید که بهتر است به جای عبارت برنامه نویسی از نشانه گذاری استفاده کنیم).
Text: این عبارت نشان می دهد که یک برنامه به زبان HTML یک فایل متنی ساده بیشتر نیست. (در واقع مرورگر ها این فایل ساده متنی را خوانده و با توجه به دستورات داخل آن، آن را به عناصر مختلف همچون عکس ها، پاراگراف ها و... تبدیل می کنند).

Hyper: در مقابل liner قرار دارد. جالب است بدانید زبان های برنامه نویسی را به دو دسته تقسیم می کنند:

- **Liner:** به زبان هایی همچون C++, Pascal, Delphi و... که باید دستورات به ترتیبی که نوشته شده اند و به صورت خط به خط اجرا شوند و تا خطی از برنامه اجرا نشده خط بعدی اجرا نمی شود و در واقع الگوی خطی دارند زبان های Liner می گویند. وجود خطا در هر خط از برنامه هایی که به یکی از زبان های liner نوشته می شوند موجب توقف اجرای آن تا تصحیح آن خطا خواهد شد.

- **Hyper:** در مقابل زبان هایی هستند که الگوی خطی ندارند و خطا در یک خط از برنامه موجب توقف کل برنامه نمی شود. به این زبان ها، زبان های Hyper گفته می شود.

Markup: HTML بر خلاف بسیاری از زبان های برنامه نویسی فاقد ساختار(دستور)هایی همچون ساختار شرطی، حلقه، توابع و... است. در عوض این زبان از ساختار ساده ای به نام Markup Tag یا به اختصار Tag (برچسب) تشکیل شده است. در واقع HTML یک نوع دستور بیشتر ندارد و آن ها همین Tag ها هستند.

ساختار Tag ها

Tag ها از سه قسمت تشکیل می شوند:

۱. یک علامت کوچکتر (<)
۲. نام تگ
۳. یک علامت بزرگتر (>)

<tagname>

تگ ها معمولاً به صورت جفتی به کار می روند و ساختار جدیدی به نام عنصر^۱ را تشکیل می دهند. به مثال زیر دقت نمایید:

<tagname>content</tagname>

¹ element

همانطور که در مثال بالا می‌بینید یک عنصر از سه قسمت اصلی تشکیل می‌شود. تگ آغازین^۱، محتوای تگ (که می‌تواند یک متن ساده، یک عنصر دیگر و یا خالی باشد) و تگ پایانی^۲.

به ساختار تگ پایانی توجه کنید که تفاوت آن با تگ آغازین علامت / ی است که قبل از نام تگ در تگ پایانی می‌آید. اجازه بدهید با اولین دستور ساده HTML که متنی را به صورت ضخیم (bold) نمایش می‌دهد آشنا شویم. به مثال زیر دقت کنید:

```
<b> HTML Learning </b>
```

کد ۱-۳

به این نکته دقت داشته باشید که هر تگی که باز می‌شود باید در جایی بسته شود. (البته استثنائاتی هم وجود دارد.)

ساختار اصلی یک صفحه وب

هر صفحه وب (HTML) با تگ `<html>` شروع و به تگ بسته `</html>` ختم می‌شود. در واقع با این عنصر به مرورگر می‌گوییم که اولاً با یک صفحه از نوع HTML روبروست ثانیاً صفحه با تگ باز `<html>` شروع و با تگ پایانی `</html>` پایان می‌یابد.

در داخل عنصر `html` دو عنصر اصلی صفحه وجود دارد:

عنصر `<head>`

این عنصر که قسمت `head` (سر) صفحه را مشخص می‌کند حاوی اطلاعاتی در مورد صفحه است. به عنوان مثال می‌تواند شامل عنوان و توضیحاتی در مورد آنچه در قسمت بدنه آمده است باشد. این قسمت از تگ `<head>` و تگ بسته `</head>` و هر چیزی بین آن‌ها تشکیل می‌شود.

عنصر `<body>`

مشخص کننده ی بدنه صفحه است و در واقع حاوی اطلاعاتی است که قرار است در پنجره مرورگر نمایش داده شود. این قسمت از تگ `<body>` و تگ بسته `</body>` و هر چیزی بین آن‌ها تشکیل می‌شود.

معمولاً آنچه که در قسمت `head` صفحه قرار می‌گیرد در صفحه نمایش داده نمی‌شود اما هر آنچه که در قسمت `body` قرار می‌گیرد در صفحه نمایش داده خواهد شد. (البته خیلی از مرورگرها از این قانون پیروی نکرده و رفتارهای متفاوتی از خود نشان می‌دهند.)

سه عنصر `<html>`، `<head>` و `<body>` با هم ساختار اصلی یک صفحه وب را تشکیل می‌دهند. مثال زیر ساختار اصلی یک صفحه را نمایش می‌دهد:

¹ Start tag/opening tag

² End tag/closing tag

```
<html>
  <head>
    <title>Title of the Page</title>
  </head>
  <body>
    <p>This is a Paragraph</p>
  </body>
</html>
```

کد ۲-۳

ذکر این نکته اهمیت دارد که وقتی عناصر شامل عناصر دیگری باشند، توی هم قرار گرفتن آن ها باید به صورت مناسب انجام شود، یعنی هر عنصر به طور کامل درون عنصر پدرش قرار گیرد. هر وقت که از یک تگ بسته استفاده می کنید، این تگ بسته، باید وابسته به آخرین تگ بازی باشد که هنوز بسته نشده. به عبارتی دیگر، اول تگ A را باز کنید، سپس تگ B را باز کنید، سپس تگ B را ببندید و در آخر تگ A را ببندید. به عنوان مثال:

```
<p> This paragraph contains some <em> emphasized text. </em></p>
```

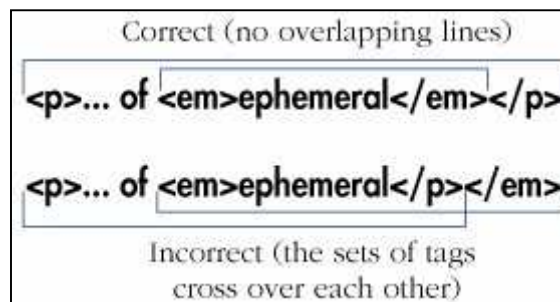
کد ۳-۳

اما مثال زیر نادرست است زیرا تگ داخل عنصر P قرار ندارد:

```
<p> This paragraph contains some <em> emphasized text. </p> </em>
```

کد ۴-۳

برای تشخیص قرار گیری صحیح عناصر در داخل یکدیگر کافی است تگ های باز و بسته مربوط به هم را با یک خط به هم وصل نمایید. حال چنانچه خطوط همدیگر را قطع کنند کد شما نادرست و در صورتی که خطوط همدیگر را قطع نکنند نحوه قرار گیری صحیح خواهد بود. به تصویر زیر نگاه کنید:



شکل ۱-۳

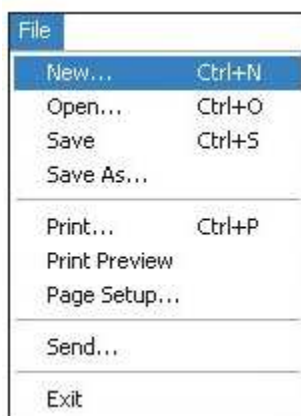
یکی از عناصری که مخصوص قرار گرفتن در قسمت head صفحه می باشد عنصر <title> است. متنی که داخل این عنصر قرار می گیرد عنوان صفحه حاصل را مشخص می کند. هر صفحه وب حتما باید یک و تنها یک تگ <title> داشته باشد. دقت کنید که متنی که انتخاب می کنید همان چیزی است که در لیست نتایج موتور های جستجو نشان داده می شود. پس در انتخاب آن و مرتبط بودن با صفحه وب دقت کنید !

ایجاد و ذخیره اولین صفحه وب

اجازه بدهید قبل از بررسی ادامه بحث ، حال که با ساختار اساسی یک صفحه وب آشنا شدیم نحوه ایجاد و ذخیره آن را نیز بیان کنیم. شاید تا اینجای کار این سوال برایتان مطرح شده باشد که دستورات HTML را باید داخل چه برنامه ای نوشت و آن را ذخیره کرد؟

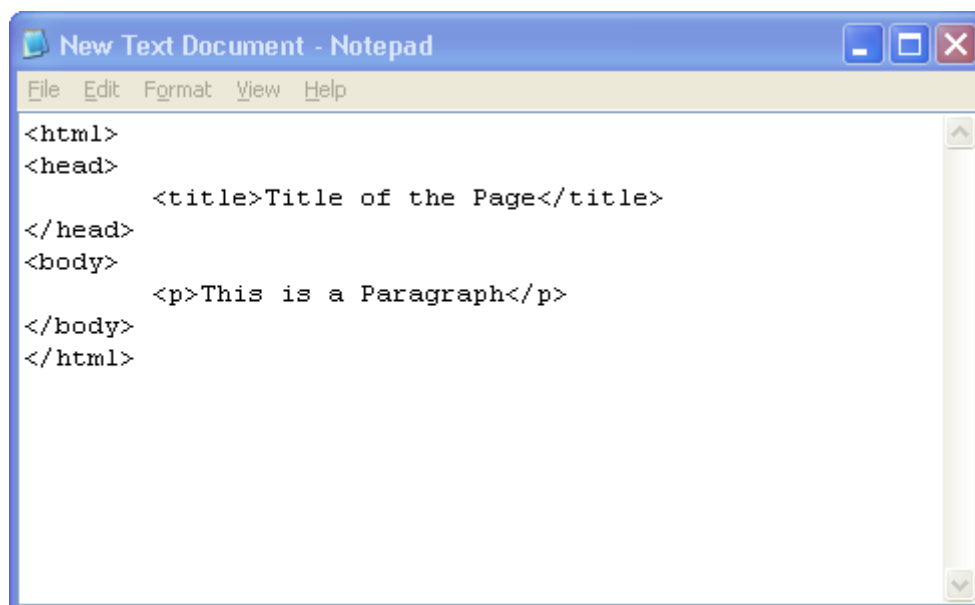
برای ایجاد یک صفحه وب به هیچ ابزار خاصی نیاز نیست ما می‌توانیم از هر یک از برنامه های واژه پرداز همچون Notepad، Wordpad، Textedit و حتی Microsoft Word استفاده نمود هر چند نرم افزار های ویژه و قدرتمندی همچون Adobe Dreamweaver و Microsoft Web Expression برای این کار وجود دارد.

برای ایجاد یک صفحه وب کافی دستورات مورد نظرتان را در یکی از برنامه های فوق تایپ کنید. برای این کار ابتدا باید یک سند جدید ایجاد کنید. تصویر زیر نحوه ایجاد یک سند جدید در برنامه Notepad را نشان می‌دهد:



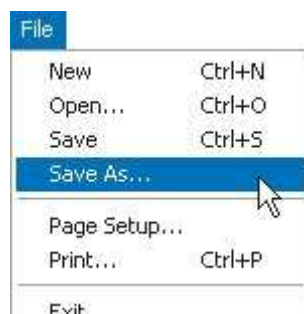
شکل ۳-۲

حال دستورات و ساختار اصلی صفحه را به صورت زیر داخل سند جدید قرار می‌دهیم:



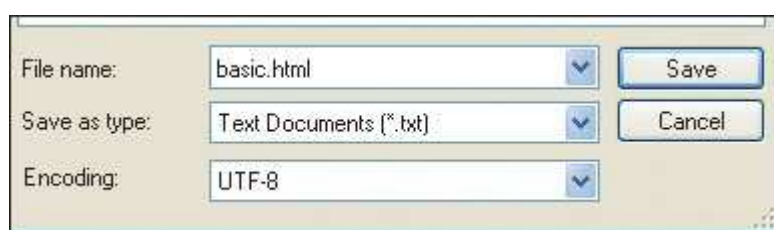
شکل ۳-۳

حال کافی است از منوی file با انتخاب گزینه Save As به صورت زیر اقدام به ذخیره آن نمایید:



شکل ۳-۴

اکنون در پنجره باز شده از قسمت File نامی برای صفحه انتخاب کنید. به دنبال آن یک نقطه قرار داده و در ادامه پسوند آن را برابر یکی از مقادیر html یا htm قرار دهید. ضمناً دقت داشته باشید که گزینه Save as type را برابر Text Document و گزینه Encoding را هم برابر UTF-8 قرار دهید. به صورت زیر:



شکل ۳-۵

حال کافی است صفحه ایجاد شده را در مرورگر خود باز کنید. حاصل به صورت زیر خواهد بود:



شکل ۳-۶

صفت ها در HTML

ما به وسیله صفت ها در HTML می‌توانیم کنترل بیشتری بر تگ ها داشته باشیم. بسیاری از تگ ها صفت های منحصر به خود را دارند که به این صفت ها، **صفت های اختصاصی** می‌گویند اما بعضی از صفات نیز بین تگ ها مشترک است و در همه آنها یک کار را انجام می‌دهند به این صفت ها، **صفت های عمومی** است. صفت ها همیشه در تگ آغازین به کار می‌روند و از دو قسمت اصلی تشکیل می‌شوند:

Name: نام صفتی که می‌خواهید کنترل کنید. مثلاً صفت `align` برای تگ `p` نحوه چینش متن پاراگراف را کنترل می‌کند.

Value: مقداری است که برای صفت تعیین می‌شود. مثلاً مقدار `right` برای صفت `align` باعث راست چین شدن متن پاراگراف می‌شود.

Value ها همیشه باید داخل کوتیشن ها (`single` یا `double`) قرار بگیرند و با علامت `=` به نام صفت منتسب شوند. عناصر می‌توانند چندین صفت داشته باشند اما نباید صفت ها، مشابه باشند! شکل کلی تعریف صفات برای عناصر به این صورت است:

```
<tagname attribute1='value' attribute2='value'...> content</tagname>
```

کد ۳-۵

مثال زیر پاراگراف را راست چین می‌کند:

```
<p align='right'>This is a paragraph</p>
```

کد ۳-۶

همانطور که اشاره شد بعضی از صفت ها که به عنوان صفت های عمومی شناخته می‌شوند در تمام تگ ها کار یکسانی را انجام می‌دهند. در زیر برخی از این صفات و وظیفه هر یک را مشاهده می‌کنید:

class: از این صفت برای نسبت دادن کلاسی که به زبان CSS نوشته شده است به یک عنصر استفاده می‌شود.

dir: جهت گیری متن داخل یک عنصر را کنترل می‌کند. این صفت دو مقدار `rtl` برای جهت گیری از راست به چپ و `ltr` برای جهت گیری از چپ به راست را به عنوان مقدار می‌پذیرد. مقدار پیش فرض این صفت `ltr` است.

id: از این صفت برای تعیین یک شناسه برای یک عنصر استفاده می‌شود. این شناسه می‌بایست در صفحه یکتا باشد. از این صفت همچنین برای نسبت دادن یک گزینشگر ID به یک عنصر استفاده می‌شود.

lang: از این صفت معمولاً برای تعیین زبان اصلی به کار رفته در صفحه و برای تگ `<html>` استفاده می‌شود. شما می‌توانید این صفت را برای هر نوع عنصر دیگری که زبان آن غیر از آن چیزی است که در تگ `<html>` تعریف شده است نیز استفاده کنید. مقداری که این صفت می‌پذیرد یکی از کدهای زبانی مانند `en,fa,fr,...` خواهد بود.

style: از این صفت برای تعیین یکسری قواعد و مقادیرشان که در زبان CSS تعریف می‌شوند برای یک عنصر می‌توان استفاده نمود.

title: این صفت مقداری از نوع رشته ساده را دریافت کرده و آن را به صورت `tooltip` هنگامی که نشانگر موس بر روی عنصر قرار گیرد نمایش می‌دهد.

contenteditable: این صفت که یکی از مقادیر `true` و `false` را اختیار می‌کند مشخص می‌کند که آیا عنصر مورد نظر قابل ویرایش توسط کاربر باشد یا خیر؟ `true` به معنی قابل ویرایش بودن عنصر است.



hidden: این صفت برای مخفی کردن عناصر استفاده می‌شود و یکی از دو مقدار true و false که true به معنی مخفی شدن عنصر است را به عنوان مقدار می‌پذیرد. عناصر hidden در صفحه نمایش داده نمی‌شوند. از این صفت می‌توان برای مخفی نگه داشتن یک عنصر از دید کاربر تا زمان برقراری یک شرط استفاده نمود. در این حالت جاوااسکریپت می‌تواند با حذف یا تغییر این صفت عنصر را مجدداً نمایش دهد.

مشاهده کدهای منبع^۱ صفحات

صفحات وب دارای مزیتی هستند و آن مشاهده کدهای HTML در مرورگر است که شما به راحتی می‌توانید تگهای عناصر تشکیل دهنده آن صفحه را ببینید و با نحوه قرار گرفتن و تکنیکهای بکار گرفته شده آشنا شوید. برای دیدن سورس کد یا همان کد تشکیل دهنده صفحه وب می‌بایست ابتدا به منوی view در مرورگر مراجعه کرده و بر روی گزینه Source کلیک می‌کنید که بطور معمول برنامه Notepad باز شده و تگهای HTML را نشان می‌دهد که در حال حاضر فهمیدن آنها برای شما مشکل است ولی در آینده نزدیک هیچ مشکلی برای درک کدها نخواهید داشت. در فصل‌های بعدی با انواع تگ‌ها، صفات و کاربردهای آنها آشنا خواهیم شد.

¹ Sourc code

فصل چهارم

آشنایی با تگ های قالب بندی متن

در این فصل با ساده ترین تگ های موجود در HTML آشنا خواهیم شد. اکثر این تگ ها برای کنترل نحوه نمایش متن ها استفاده می شوند که از این رو به آن ها تگ های قالب بندی متن گفته می شود.

آشنایی با انواع تگ های قالب بندی متن

چندین تگ مخصوص قالب بندی متن های نمایشی در صفحه وجود دارد که در ادامه به بررسی هریک از آن ها و صفات و مقادیرشان خواهیم پرداخت:

تگ



متن داخل یک عنصر b در صفحه به صورت ضخیم^۱ نمایش داده خواهد شد. در کد زیر عبارت bold در صفحه به صورت ضخیم نمایش داده خواهد شد:

The following word uses a **bold** typeface.

کد ۱-۴

تگ <i>



از این تگ برای مورب^۲ کردن متن استفاده می شود. کلمه italic در کد زیر در صفحه به صورت مورب نمایش داده خواهد شد.

The following word uses a *italic* typeface.

کد ۲-۴

تگ <u>



این عنصر محتوای خود را به صورت زیر خط نمایش می دهد. این عنصر جزء عناصر منسوخ شده به شمار می رود. با این حال همه مرورگر ها از آن به خوبی پشتیبانی می کنند.

تگ های <s> و <strike>



این دو تگ محتوای خود را به صورت خط خورده نشان می دهند. در واقع یک خط نازک بر روی متن ایجاد می کنند.

تگ <sup>



محتوای این تگ به صورت بالانویس^۳ نشان داده خواهد شد. بالانویس ها معمولاً به اندازه نصف ارتفاع یک کاراکتر بالاتر از بقیه کاراکتر ها قرار می گیرند و کمی کوچکتر از متون اطرافشان هستند. به مثال زیر دقت کنید:

Written on the 31st February.

کد ۳-۴

تگ <sub>



محتوای این عنصر به صورت زیر نویس^۴ (اندیس) نشان داده خواهد شد. زیر نویس ها معمولاً به اندازه نصف ارتفاع یک کاراکتر پایین تر از بقیه کاراکتر ها قرار می گیرند و کمی کوچکتر از متون اطرافشان هستند.

^۱ bold

^۲ italic

^۳ superscript

^۴ subscript

تگ <big>



در نسخه‌های قدیمی‌تر HTML، اندازه استاندارد برای متن وجود داشت. عنصر <big> محتوای خود را یک واحد بیشتر از متون اطرافش نمایش می‌دهد.

```
he following word should be < big > bigger < /big > than those around it.
```

کد ۴-۴

حاصل اجرای کد فوق را در شکل زیر می‌بینید:

The following word should be **bigger** than those around it.

استفاده از این عنصر به صورت تو در تو موجب اثر مضاعف خواهد شد :

```
<big>this <big>text is <big>written</big> by big</big> tag</big>
```

کد ۵-۴

حاصل این کد هم به شکل زیر است:

this text is **written** by big tag

تگ <small>



این عنصر دقیقاً عکس عنصر <big> عمل می‌کند و محتوایش را یک واحد کوچکتر از متون اطرافش نمایش می‌دهد:

```
The following word should be <small> smaller </small> than those around it.
```

کد ۶-۴

استفاده از این عنصر به صورت تو در تو اثر مضاعف خواهد داشت.

تگ‌های و <ins>

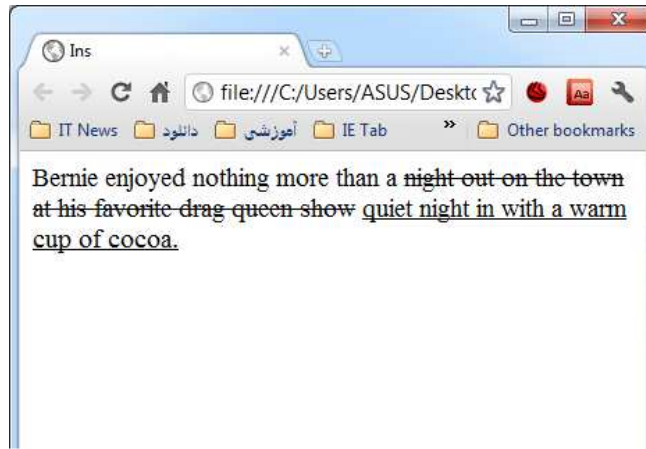


از تگ del برای مشخص کردن تغییراتی که نشان می‌دهد نویسنده صفحه، محتوایی که از قبل در صفحه موجود بوده است را حذف کرده است می‌توان استفاده کرد. این تگ در مقابل تگ ins که وظیفه مشخص کردن محتوایی که به تازگی به صفحه اضافه شده است را دارد قرار دارد. تگ del موجب نمایش خطی بر روی محتوای خود و تگ ins موجب نمایش زیر خط دار متن محتوای خود خواهد شد:

```
<p>Bernie enjoyed nothing more than a <del>night out on the town at his favorite drag queen show</del> <ins>quiet night in with a warm cup of cocoa.</ins></p>
```

کد ۷-۴

حاصل اجرای کد فوق در مرورگر به صورت زیر خواهد بود:



شکل ۴-۱

تگ های <abbr> و <acronym>



از این تگ ها برای تعیین عبارت های کامل کلماتی که به صورت مختصر و کوتاه شده در صفحه ذکر شده اند استفاده می شود. هر دو این تگ ها از صفت title برای تعیین عبارت کامل این گونه کلمات پشتیبانی می کنند. مثال زیر یک نمونه از کاربرد تگ abbr به همراه صفت title آن را نشان می دهد:

```
<p>The page had the phrase <abbr title="Please Turn Over">PTO</abbr>
written on it. If only I knew what PTO meant. So I turned the page
to see if that revealed the answer.</p>
```

کد ۴-۸

حاصل اجرای کد فوق در مرورگر به صورت زیر خواهد بود:



شکل ۴-۲

ایجاد تیتر ها با استفاده از تگ های <h>

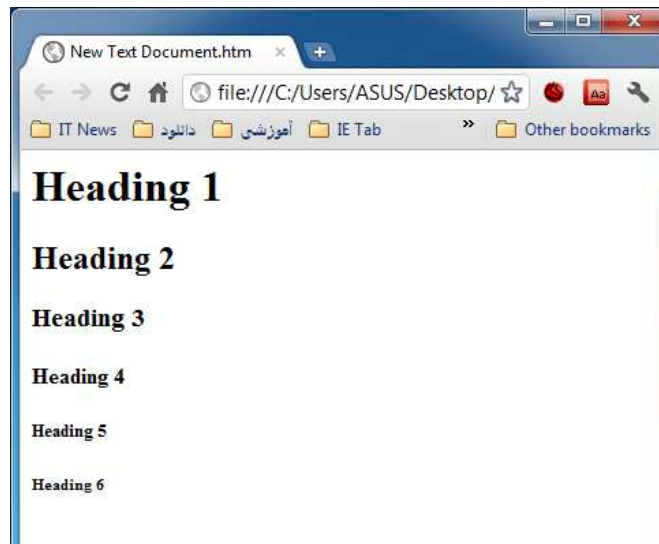


X(HTML) از ۶ سطح تیتر (Heading) ها پشتیبانی می کند. این ۶ سطح با تگ های <h1>، <h2>، <h3>، <h4>، <h5>، <h6> ایجاد می شوند. مرورگر ها عنصر <h1> را در بزرگترین و عنصر <h6> را در کوچکترین اندازه از بین این ۶ سطح نمایش می دهند. به مثال زیر دقت نمایید:

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```

کد ۴-۹

حاصل اجرای کد فوق در مرورگر به شکل زیر خواهد بود:



شکل ۴-۳

هر ۶ عنصر فوق دارای صفت های زیر هستند که در ادامه بررسی خواهیم کرد:

```
align class id style title dir lang
```

کد ۴-۱۰

align: این صفت نحوه چینش متن داخل heading را کنترل می‌کند.

left: heading در سمت چپ صفحه (یا عنصری که heading در آن قرار دارد) قرار می‌گیرد.

center: heading در وسط صفحه (یا عنصری که heading در آن قرار دارد) قرار می‌گیرد.

right: heading در سمت راست صفحه (یا عنصری که heading در آن قرار دارد) قرار می‌گیرد.

justify: متن داخل heading را از هر دو طرف تراز می‌کند.

title: این صفت که جزء صفت های عمومی محسوب می‌شود رشته ای را به عنوان مقدار می‌پذیرد و آن را به صورت tooltip نمایش می‌دهد.

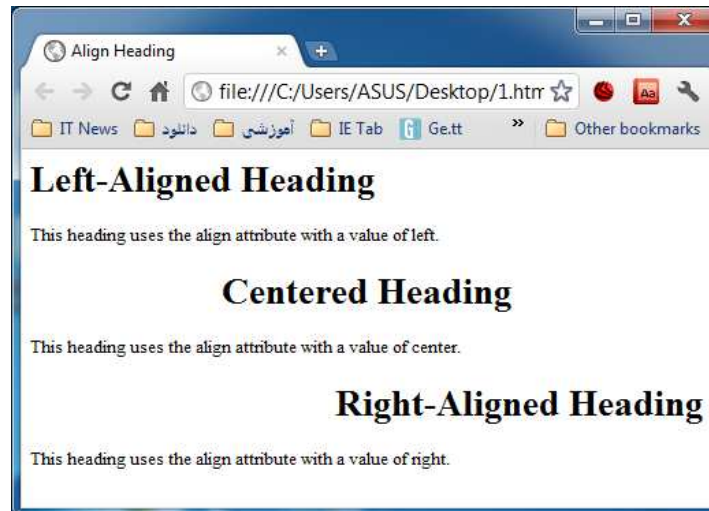
dir: این صفت جهت گیری متن داخل heading ها را کنترل می‌کند و می‌تواند یکی از مقادیر rtl برای جهت گیری راست به چپ و ltr برای جهت گیری چپ به راست را می‌پذیرد.

به تکه کد زیر دقت کنید:

```
<h1 align="left"> Left-Aligned Heading </h1>
<p> This heading uses the align attribute with a value of left. </p>
<h1 align="center"> Centered Heading </h1>
<p> This heading uses the align attribute with a value of center. </p>
<h1 align="right"> Right-Aligned Heading </h1>
<p> This heading uses the align attribute with a value of right. </p>
```

کد ۴-۱۱

شکل زیر حاصل اجرای کد فوق را نشان می‌دهد:



شکل ۴-۴

ایجاد پاراگراف ها با استفاده از تگ <p>



از تگ P برای ایجاد پاراگراف ها در صفحه استفاده می‌شود. معمولا همیشه پاراگراف ها از خط جدید شروع می‌شوند و یک فضای خالی قبل و بعد از خود ایجاد می‌کنند. کد زیر سه پاراگراف ایجاد می‌کند:

```
<p> Here is a paragraph of text. </p>
<p> Here is a second paragraph of text. </p>
<p> Here is a third paragraph of text. </p>
```

کد ۴-۱۲

این تگ از صفت های عمومی زیر پشتیبانی می‌کند:

align class id style title dir lang

تگ
 برای ایجاد شکست خط



هر آنچه که بعد از عنصر
 قرار می‌گیرد از یک خط جدید شروع می‌شود. این عنصر جزء عناصر تهی^۱ بوده و به تگ پایانی احتیاج ندارد. شما از این عنصر می‌توانید به شکل
 نیز استفاده کنید که مخصوص نسخه های قدیمی HTML بوده اما در نسخه XHTML می‌بایست بعد از
 یک فاصله خالی، در ادامه یک / و در انتها یک علامت > قرار گیرد.

تگ <no>



این تگ به نوعی درمقابل تگ
 قرار دارد. به این دلیل که متن داخل این تگ به هیچ عنوان شکسته نخواهد شد مگر اینکه تگ
 داخل آن قرار بگیرد. استفاده از این تگ باید با احتیاط انجام شود زیرا ممکن است موجب ایجاد scroll افقی در پنجره مرورگر شود که معمولا این حالتی عذاب آور برای کاربر صفحه خواهد بود!

¹ Empty element/tag

تگ‌های `<pre>` و `<code>`

گاهی اوقات می‌خواهیم متنمان دقیقاً به همان شکلی که نوشته می‌شود در صفحه نمایش داده شود و از فاصله‌های خالی، فاصله‌های Tab و شکست‌های خط چشم پوشی نشود. هر آنچه که بین تگ `<pre>` و `</pre>` قرار می‌گیرد دقیقاً به همان شکلی که در source صفحه نوشته شده است در صفحه نشان داده خواهد شد. این تگ معمولاً به همراه تگی با نام `<code>` که متن داخل آن به یکی از فونت‌های از نوع monospace همانند courier new نشان داده می‌شود به کار می‌رود. (فونت‌های نوع monospace فونت‌هایی هستند که هر کاراکتر آن‌ها به یک اندازه فضا اشغال می‌کنند.)

بیشترین استفاده از این تگ‌ها در نمایش کدهای کامپیوتری و برنامه نویسی است. در مثال زیر نمونه کدی به زبان جاوااسکریپت در داخل تگ `<pre>` و `<code>` قرار گرفته است که در صفحه دقیقاً به همان شکل نمایش داده خواهد شد:

```
<code>
  <pre>
    function testFunction(strText){
      alert (strText)
    }
  </pre>
</code>
```

کد ۴-۱۳

تگ `<hr />`

از این تگ برای ایجاد خطوط افقی در صفحات استفاده می‌شود. این تگ جز تگ‌های تهی بوده و باید دقیقاً به همین شکل استفاده شود.

Width: با این صفت می‌توان عرض عناصر موجود در صفحه را کنترل کرد. به دو طریق می‌توان به این گونه صفات مقداردهی کرد. یکی بر حسب px و دیگری بر حسب درصد که نسبت به پهناي کل صفحه (یا عنصری که hr در آن قرار دارد) سنجیده می‌شود.

Color: این قاعده رنگ خط را مشخص می‌کند. صفاتی از این دست در HTML که یک رنگ را به عنوان مقدار می‌پذیرند را می‌توان به سه روش مقدار دهی نمود:

① **استفاده از نام رنگ‌ها:** ما از نام ۱۶ رنگ استاندارد می‌توانیم برای مقداردهی به صفت‌ها استفاده کنیم. این رنگ‌ها به شرح زیر هستند:

Black, silver, gray, white, maroon, red, purple, fuchsia, green, lime, olive, yellow, navy, blue, teal, aqua

② **استفاده از تابع rgb:** در HTML تابعی به نام rgb وجود دارد که به ترتیب سه رنگ قرمز، سبز و آبی را به عنوان مقدار می‌پذیرد. این مقادیر یا اعداد صحیح هستند یا به صورت درصد بیان می‌شوند. به طور مثال: rgb(0,160,255) یا rgb(0%,63%,100%). در اولین عبارت، عدد 255 معادل 100% است. در مد رنگ RGB محدوده اعداد صحیح بین ۰ تا ۲۵۵ است.

۷ استفاده از مبنای هگزادسیمال (۱۶) رنگ ها: رنگ ها را می توان بر مبنای عبارت متناظر آن ها در مبنای ۱۶ به کار برد. در این عبارت می توان سه عدد در مبنای ۱۶ را نوشت. عدد اول معرف رنگ قرمز، عدد دوم معرف رنگ سبز و عدد سوم معرف رنگ آبی است. دقت داشته باشید که این اعداد می توانند از دو حرف تشکیل شده باشند زیرا اعداد در مبنای ۱۶ از حروف A تا F نیز می توانند تشکیل شوند فرم کلی این نوع عبارت ها به شکل #rrggbb است. برای مثال عبارت #000000 معرف رنگ سیاه و #0000ff معرف رنگ آبی است.

Size: این صفت ارتفاع یک خط را کنترل می کند و می تواند یک عدد بر حسب px را به عنوان مقدار می پذیرد.

noshade: خط هایی که در صفحه ایجاد می شوند به صورت پیش فرض دارای خطی در زیر خود به عنوان سایه هستند. ما می توانیم از این صفت بولین برای حذف این سایه استفاده کنیم.

صفت های Boolean به صفت هایی گفته می شود که در نسخه های قدیمی HTML (نسخه های قبل از XHTML) مقداری را نمی پذیرند و ذکر نام آن ها برای استفاده از آن ها کافی است مثلا: noshade. اما در نسخه XHTML می بایست این گونه صفت ها را نیز مقدار دهی کنیم و مقدار آن ها نام خود آن ها خواهد بود مثلا: noshade="noshade".

align: محل قرار گیری خط در صفحه را کنترل می کند و می تواند یکی از موارد right، center و left را به عنوان مقدار بپذیرد.

تگ <blink>

متن داخل این عنصر در مرورگر به صورت چشمک زن در خواهد آمد. این تگ تنها در مرورگر های Firefox، Google Chrome و opera پشتیبانی می شود و IE از آن پشتیبانی نمی کند.

تگ <center>

از این تگ برای وسط چین کردن هر نوع عنصری استفاده می شود. برای این کار کافی است تگ باز <center> را قبل و تگ بسته </center> را بعد از عنصر مورد نظرتان را قرار دهید.

تگ

از این تگ برای تغییر رنگ، تغییر اندازه و تغییر فونت متون و سایر عناصر متنی موجود در HTML استفاده می شود. این تگ که جزء تگ های درون خطی محسوب می شود از صفت های زیر پشتیبانی می کند:

face: از این صفت برای مشخص کردن فونت خاصی استفاده می شود. یعنی اگر بخواهیم متنی را با فونت خاصی نمایش دهیم از تگ همراه با این صفت استفاده می کنیم. این صفت لیستی از فونت ها که با (کاما) از هم جدا شده اند را به عنوان مقدار می پذیرد. اگر اولین فونت بر روی سیستم کاربر نصب نبود، فونت دوم، فونت دوم نصب نبود فونت سوم و... بر روی متن اعمال می شود.

Size: از این صفت برای تعیین اندازه متون استفاده می‌شود. به دو شکل می‌توان به این صفت مقدار داد:

▼ **مقادیر مطلق:** مقادیری بین 1 تا 7 که در این مورد هر اندازه از اندازه قبلی خود بزرگتر خواهد بود.

▼ **مقادیر نسبی:** مقادیری بین -7 تا +7 که در این صورت اندازه فونت نسبت به اندازه فونت مشخص شده در تگ `<basefont />` سنجیده خواهد شد. (در مورد این تگ در ادامه خواهید خواند.) اگر عدد مشخص شده مثبت باشد به این معنی است که به همان تعداد به اندازه مشخص شده در `<basefont />` اضافه و اگر عدد مشخص شده منفی باشد به این معنی است که به همان تعداد از اندازه مشخص شده در `<basefont />` کم شود و سپس بر روی متن اعمال گردد.

Color: از این صفت برای تعیین رنگ یک متن استفاده می‌شود. این صفت مقداری از انواع رنگ‌ها را می‌پذیرد و رنگ متن را مشخص می‌کند.

تگ `<basefont>`

از این تگ تهی که حتما باید در قسمت head استفاده شود برای تعیین اندازه و نوع و رنگ font پایه صفحه استفاده می‌شود. بیشترین استفاده از این تگ برای تعیین اندازه font پایه صفحه است تا تمام اندازه‌های نسبی صفحه تگ‌های `` داخل صفحه نسبت به آن سنجیده شود.

در صورتی که اندازه خاصی در تگ `<basefont>` تعیین نشود اندازه به صورت پیش فرض 3 در نظر گرفته می‌شود. این تگ از تمام صفت‌های تگ `` (یعنی size, face, color) با همان نوع مقادیر پشتیبانی می‌کند جز اینکه به صفت size آن فقط می‌توان مقادیر مطلق داد و مقادیر نسبی مجاز نیستند.

تگ `<marquee>`

از این تگ برای به حرکت درآوردن متون، عکس‌ها و سایر عناصر استفاده می‌شود. صفت‌های این تگ به شرح زیر است:

Width: عرض محدوده marquee را بر حسب درصد یا پیکسل مشخص می‌کند.

Height: ارتفاع محدوده marquee را بر حسب درصد یا پیکسل مشخص می‌کند.

Bgcolor: رنگ پس‌زمینه محدوده marquee را مشخص می‌کند. به سه روشی که قبلاً به آن اشاره شده است می‌توان به این صفت مقدار داد.

Behavior: نحوه رفتار یا حرکت محتویات marquee را مشخص می‌کند که می‌تواند یکی از مقادیر زیر باشد:

▼ **Scroll:** متن از یک طرف وارد و از طرف دیگر خارج می‌شود.

▼ **Slide:** متن از یک طرف وارد و در طرف دیگر می‌ایستد

▼ **Alternate:** متن (محتویات) داخل marquee یک حرکت رفت و برگشتی در سطح آن خواهد داشت.

Scrollamount: اگر حرکت متن بوسیله تگ `marquee` را مانند قدم زدن در نظر بگیریم این صفت فاصله بین هر قدم را کنترل می‌کند. این صفت مقداری عددی بر حسب `px` را می‌پذیرد

Scrolldelay: این صفت تاخیر زمانی بین هر قدم را بر حسب میلی ثانیه مشخص می‌کند. مثلاً مقدار ۱۰۰۰ برای این صفت به معنی تاخیر یک ثانیه ای برای هر قدم خواهد بود.

Loop: تعداد تکرار حرکت متن را کنترل می‌کند و می‌تواند یک عدد صحیح باشد. در صورتی که بخواهید حرکت بی نهایت بار تکرار شود آن را برابر با یک عدد منفی (مثلاً -1) یا عبارت `infinite` قرار بدهید.

Direction: جهت حرکت محتویات `marquee` را مشخص می‌کند و می‌تواند یکی از مقادیر زیر باشد:

Up: حرکت به سمت بالا

Down: حرکت به سمت پایین

Right: حرکت به سمت راست

Left: حرکت به سمت چپ

hspace: میزان فاصله طرف های راست و چپ محدوده `marquee` نسبت به عناصر اطراف را بر حسب `px` یا در صد کنترل می‌کند.

vspace: میزان فاصله طرف های بالا و پایین محدوده `marquee` نسبت به عناصر اطراف را بر حسب `px` یا در صد کنترل می‌کند.

truespeed: معمولاً مرورگرها مقادیر کمتر از ۶۰ برای صفت `scrolldelay` را در نظر نمی‌گیرند و آن را به ۶۰ گرد کرده و سپس اعمال می‌کنند. ما می‌توانیم با به کارگیری صفت `truespeed` که جزء صفت های `boolean` نیز هست از این تبدیل جلوگیری نماییم. در واقع با استفاده از این صفت می‌توانیم مرورگرها را مجبور کنیم تا دقیقاً همان عددی را که در صفت `scrolldelay` مشخص کرده ایم در حرکت متن تاثیر بدهند.

عناصر بلاکی^۱ و درون خطی^۲

یکی از مفاهیم کلیدی HTML که درک درست از آن کمک شایانی به درک دیگر مفاهیم می‌کند تقسیم بندی عناصر به دو گروه بلاکی و درون خطی است.

عناصر بلاکی عناصری هستند که همیشه از یک خط جدید شروع می‌شوند و می‌توانند حاوی عناصر بلاکی و درون خطی دیگر باشند. در واقع این گونه عناصر حالت بلاکی داشته و همیشه در ابتدا و انتهای خود یک خط خالی ایجاد می‌کنند. عناصری همچون `p, h1...h6, ol, ul, pre, hr` از این دست هستند.

¹ Block level

² Inline level

در مقابل عناصر درون‌خطی عناصری هستند که معمولاً برای نمایش یک متن ساده استفاده می‌شوند و هیچ‌گاه از خط جدید شروع نمی‌شوند. این گونه عناصر فقط می‌توانند شامل عناصر درون‌خطی دیگر باشند (و نمی‌توانند حاوی عناصر بلاکی دیگر باشند) از این نوع عناصر می‌توان به عناصری همچون `sub`، `big`، `small`، `a`، `img` اشاره کرد.

البته نوع سوم از عناصر نیز وجود دارند که اصلاً در صفحه نمایش داده نمی‌شوند و تنها یکسری اطلاعات در مورد صفحه را فراهم می‌کنند. برای مثال: تگ `<style>` برای تعریف قواعد CSS، تگ `<meta>` برای اطلاعاتی در مورد صفحه و تگ `<head>` برای مشخص کردن قسمت سر صفحه.

فصل پنجم

کار با لینک ها

بدون شک یکی از ویژگی هایی که وب را از سایر ابزار و رسانه ها متمایز می کند این است که در یک صفحه وب می توان با کلیک بر روی یک لینک به سایر صفحات و منابع اطلاعاتی موجود بر روی اینترنت دسترسی داشت.

موقعی که شما از لینک ها برای پیوند دادن یک صفحه به سایر صفحات سایتتان استفاده می کنید به این لینک ها، لینک های نسبی (داخلی)^۱ و زمانی که از آن ها برای پیوند به صفحات موجود در سایت های دیگر استفاده می کنید به آن ها لینک های خارجی^۲ گفته می شود. به طور مشابه آدرس هایی که در لینک ها استفاده می شود نیز به دو نوع نسبی و مطلق تقسیم بندی می شوند.

لینک ها به انواع دیگری همچون لنگری، ایمیلی و دانلود نیز دسته بندی می شوند. ما در این فصل به بررسی انواع لینک ها، آدرس ها و نحوه ایجاد هر یک از آن ها خواهیم پرداخت.

^۱ Internal link

^۲ External link

تگ <a>

از تگ a برای ایجاد لینک ها استفاده می شود. محتوای یک عنصر a می تواند یک متن ساده، یک عکس یا ترکیبی از این دو باشد. در مثال زیر لینکی که با کلیک بر روی آن به سایت گوگل هدایت خواهید شد را مشاهده می کنید:

```
<a href="http://google.com" target="_blank" >Click Me</a>
```

کد ۵-۱

در ادامه به بررسی مهمترین صفت های تگ a برای کنترل جزئیات لینک ها می پردازیم:

accesskey: از این صفت می توان برای تعریف کردن یک کاراکتر از صفحه کلید به عنوان میانبر^۱ برای دسترسی به یک لینک استفاده نمود. نحوه دسترسی و استفاده از یک لینک بر اساس میانبر آن در مرورگر های مختلف متفاوت است. در IE از کلید های ترکیبی **Alt+accesskey**، در Firefox و Google Chrome از کلید های ترکیبی **Alt+Shift+accesskey** و در opera از کلید های ترکیبی **Shift+Esc** برای دسترسی به لینکی که میانبری برای آن تعریف شده است استفاده می شود.

href: این صفت آدرسی که لینک به آن اشاره می کند را مشخص می کند. این آدرس می تواند آدرس صفحه ای در همان مسیر، صفحه ای مربوط به سایتی دیگر، مکانی در همان صفحه جاری و یا آدرس هر نوع فایل دیگری باشد.

hreflang: زبان به کار رفته در منبعی که لینک به آن اشاره می کند را مشخص می کند. این صفت یکی از کدهای زبانی مانند en، fa، fr و... می تواند باشد.

name: نامی برای لینک مشخص می کند. این نام باید در صفحه یکتا باشد. نامی که انتخاب می شود باید با یکی از حروف A-Z یا a-z شروع شده و در ادامه می تواند یکی از حروف A-Z یا a-z، اعداد 0-9، -، _، یا. (نقطه) باشد.

rel: این صفت که مخفف عبارت relationship است رابطه صفحه مقصد با صفحه مبدا لینک را مشخص می کند.

rev: این صفت که مخفف عبارت reverse است عکس صفت بالا عمل می کند و نوع رابطه صفحه مبدا با مقصد لینک را مشخص می کند.

مقادیری که صفت های rel و rev می پذیرند بدین شرح است:

start next, prev, Content Index, glossary, copyright, appendix, help

tabindex: بوسیله این صفت می توان ترتیب پیمایش لینک ها در صفحه که با استفاده از کلید Tab صفحه کلید انجام می شود را مشخص کنیم. مقداری که این صفت می گیرد می تواند عددی بین ۰ تا ۳۲۷۶۷ باشد.

Target: با این صفت می توان صفحه ای که لینک باید در آن باز شود را مشخص کرد و می تواند یکی از مقادیر زیر را بگیرد:

¹ shortcut

✓ **_blank**: لینک را کاملاً در صفحه ای جدید باز می‌کند.

✓ **_self**: لینک را در همان فریم (صفحه) ای که لینک قرار دارد باز می‌کند.

✓ **_top**: صفحه را در پنجره ای بدون فریم باز می‌کند (غالباً مثل **_self** عمل می‌کند)

✓ **_parent**: لینک را در **frameset** پدر تگ **frame** جاری باز می‌کند.

✓ **نام یک فریم (پنجره)**: لینک را در پنجره ای با نام مشخص باز می‌کند.

اجزای تشکیل دهنده آدرس ها

احتمالاً همانطور که تا به اینجا متوجه شدید بعضی از صفت ها مانند **href** در تگ **<a>** یا **src** در تگ **** یک URL را به عنوان مقدار می‌پذیرند. URL شامل اطلاعاتی در مورد محل فایل و اینکه مرورگر باید با آن چه کاری انجام دهد می‌باشد. هر فایل بر روی اینترنت دارای یک URL منحصر به فرد است. هر آدرس از اجزای مختلفی تشکیل می‌شود. در تصویر زیر یک آدرس اینترنتی و اجزای تشکیل دهنده آن را مشاهده می‌کنید:



شکل ۵-۱

بخش اول URL طرح^۲ (پروتکل) نامیده می‌شود. این بخش به مرورگر می‌گوید چگونه با فایلی که می‌خواهد آن را باز کند رفتار کند. از متداول ترین طرح هایی که وجود دارند می‌توان به **http**، **ftp**، **mailto** و... اشاره کرد. سایر بخش ها هم شامل مواردی همچون دامنه، پسوند، مسیر، نام فایل و پسوند آن که ما را به فایل می‌رساند می‌باشند. البته آدرس ها را به دو دسته اصلی تقسیم می‌کنند که در ادامه به بررسی جزئیات هر یک از آن ها خواهیم پرداخت.

آدرس های مطلق و نسبی

در اصل آدرس ها را به دو دسته تقسیم می‌کنند:

آدرس های مطلق^۳

به آدرس هایی که در آنها تمام اجزای یک آدرس اینترنتی همچون نام پروتکل (طرح)، جداکننده ها، عبارت **WWW**، نام سایت و دامنه ذکر می‌شود آدرس های مطلق گفته می‌شود. به عنوان مثال آدرس زیر یک آدرس مطلق است:

<http://www.site.com/page.html>

¹ Uniform Resource Locator

² scheme

³ absolute URL

آدرس های نسبی^۱

نوع دوم آدرس ها، آدرس های نسبی هستند که معمولا در آن ها از آوردن نام پروتکل، نام سایت و دامنه چشم پوشی می شود. به عنوان مثال آدرس های زیر نمونه هایی از آدرس های نسبی هستند:

```
/images/logo.gif
../../download/pascal.zip
```

کد ۲-۵

معمولا برای دسترسی به فایل هایی از داخل سرور (سایت) خودمان از آدرس های نسبی و برای دسترسی به فایل هایی در سرور (سایت) های دیگر از آدرس های مطلق استفاده می شود.

در مورد آدرس های نسبی ذکر چند نکته ضروری است:

- برای دسترسی به فایلی که در یک مسیر^۲ بالاتر از فایل اصلی قرار دارد از ../ استفاده می شود. مثلا:

```
<a href='../test.html'>Click Here</a>
```

کد ۳-۵

- برای دسترسی به فایلی که در directory اصلی سایت قرار دارد از / استفاده می شود. مثلا:

```
<a href='/home.html'>GO To HomePage</a>
```

کد ۴-۵

- برای دسترسی به فایلی که در همان directory که فایل اصلی قرار دارد از./ استفاده می شود.

```
<img src='./logo.gif' />
```

کد ۵-۵

تگ <base>



از این تگ، که جزء تگ های تهی است و حتما باید در قسمت head صفحه هم قرار بگیرد برای تعریف کردن یک آدرس پایه برای تمام لینک های نسبی صفحه استفاده میشود. همچنین با استفاده از این تگ می توان یک هدف مشخص برای باز شدن لینک ها را هم مشخص نمود. اگر تعداد زیادی از لینک ها در صفحه به آدرس مشخصی اشاره می کنند استفاده از تگ base برای جلوگیری از کد نویسی اضافه پیشنهاد می شود. این تگ دو صفت href و target را داراست که صفت href آن باید حتما از نوع آدرس های مطلق باشد.

لینک های ایمیلی

احتمالا در صفحات وب به لینک هایی که اشاره به یک آدرس ایمیل دارند برخورد کرده اید. زمانی که بر روی آن ها کلیک می کنیم برنامه ارسال ایمیل^۳ باز می شود و همه چیز را برای ارسال ایمیل به آن آدرس مهیا می کند. به این نوع لینک ها، **لینک های ایمیلی** گفته می شود.

¹ relative URL

² Directory

³ Outlook

برای ایجاد لینک های ایمیلی می بایست از سینتکس^۱ کلی زیر در عنصر a استفاده کنید:

```
<a href="mailto:name@example.com">name@example.com </a>
```

کد ۵-۶

در مثال فوق همانطور که مشاهده می کنید مقدار صفت href با عبارت mailto شروع شده و به دنبال آن کاراکتر : آمده و پس از آن آدرس ایمیلی که می خواهید ایمیل به آن ارسال شود آمده است. محتوای عنصر a مانند انواع دیگر لینک ها می تواند هر چیز دیگری نیز باشد. بنابراین مثال فوق را می توان به صورت زیر نیز نوشت:

```
<a href="mailto:name@example.com"> E-mail us </a>.
```

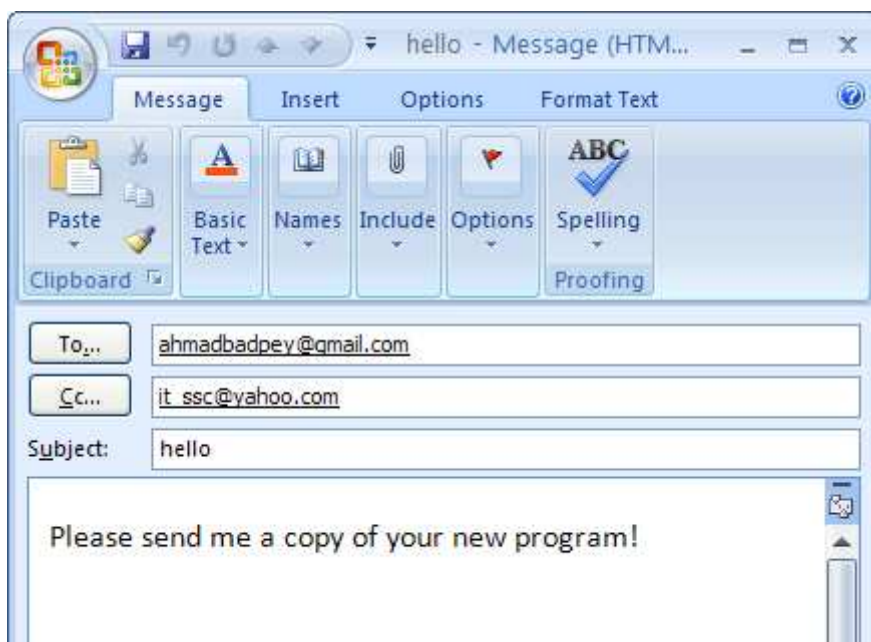
کد ۵-۷

در هنگام ایجاد لینک های ایمیلی علاوه بر مشخص کردن آدرس ایمیل دریافت کننده حتی می توان موضوع، آدرس سایر دریافت کنندگان (یعنی گزینه های CC^۲ و BCC^۳) و بخشی از متن ایمیل را نیز مشخص کرد. برای این کار کافی است کد فوق را به صورت زیر تغییر داده و از پارامترهایی همچون subject، CC و body پس از آدرس اصلی ایمیل استفاده کنید:

```
<a href="mailto:ahmadbadpey@gmail.com? subject=hello&
CC=it_ssc@yahoo.com& body=Please send me a copy of your new
program!">Email Me</a>
```

کد ۵-۸

اکنون پس از کلیک بر روی لینک فوق برنامه ارسال ایمیل که معمولاً Microsoft Office Outlook می باشد به صورت زیر و با مشخصات پیش فرض تعیین شده باز خواهد شد:



شکل ۵-۲

^۱ Syntax

^۲ Carbon Copy

^۳ Blind Carbon Copy

لینک های لنگری

برخلاف اغلب لینک ها که برای انتقال به سایر صفحات استفاده می‌شوند از لینک های لنگری برای اشاره و هدایت کاربر به بخش خاصی از همان صفحه ای که لینک در آن قرار دارد (مخصوصا در صفحات طولانی و برای صرفه جویی در وقت کاربر به منظور اسکرول کردن به بالا و پایین) می‌توان استفاده نمود. به عنوان مثال یکی از کاربرد های لینک های لنگری، در ایجاد لینک هایی با عنوان "**بازگشت به بالا**" در پایین صفحات طولانی در بسیاری از سایت هاست.

برای ایجاد لینک های لنگری می‌بایست از دو عنصر a استفاده کرد. یکی در بخشی از صفحه که می‌خواهیم کاربر به آن نقطه هدایت شود. به این لینک، **مقصد لنگر** می‌گویند. و دیگری لینکی که با کلیک بر روی آن به محل لینک مقصد هدایت می‌شویم که غالبا به آن **مبدا لنگر** می‌گویند.

در هنگام ایجاد مقصد لنگر می‌بایست به جای استفاده از صفت href در تگ a از صفت id برای نامگذاری آن استفاده کرد. ما از این نام در مبدا لنگر برای جهش به آن بخش بهره می‌بریم. حال در تگ a مبدا لنگر می‌بایست در صفت href از کاراکتر # و به دنبال آن نامی که برای مقصد انتخاب شده است استفاده کنیم.

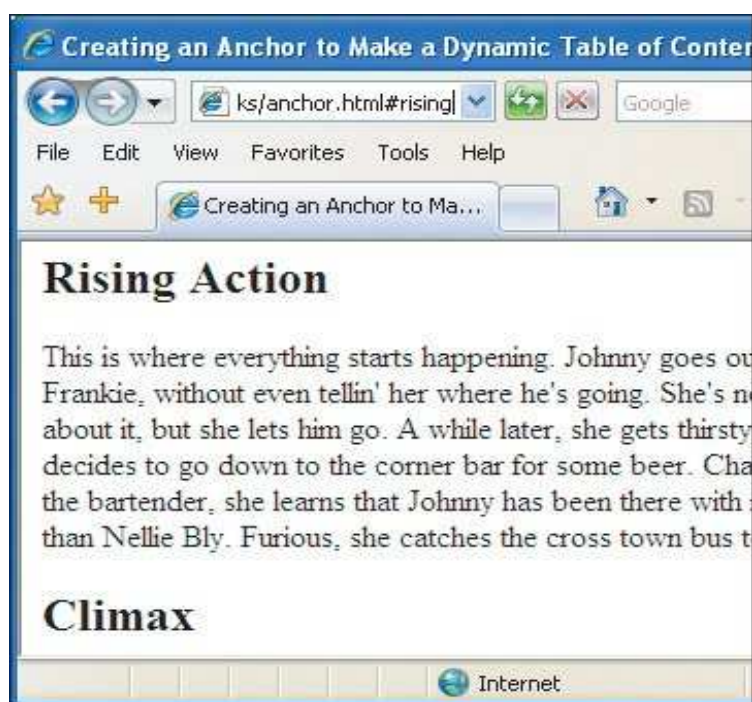

```

<body>
  <div class="toc">
    <h2>Table of Contents</h2>
    <a href="#intro">Introduction</a><br />
    <a href="#descript">Description of the Main Characters</a><br />
    <a href="#rising" >Rising Action</a><br />
    <a href="#climax" > Climax </a><br />
    <a href="#denoue" >Denouement</a>
  </div>
  <h2><a name="intro"> Introduction</a></h2>
  <p>This is a intro.If I could think of enough things to write about,
it could span a few pages , giving all the introductory information
that an introduction should introduce.
  </p>
  <h2><a name="descript" >Description of the Main Characters</a></h2>
  <p>Frankie and Johnny are the main characters. She's jealous, and
seems to have a reason to be. He's a sleaze, and will be price.</p>
  <h2><a name="rising" > Rising Action </a></h2>
  <p>This is where everything starts happening. Johnny goes out, without
Frankie, without even tellin' her where he's going. She's not crazy
about it, but she lets him go. A while later , she gets thirsty and
decides to go down to the corner bar for some beer. Chatting with the
bartender, she learns that Johnny has been there with no other than
Nellie Bly. Furious , she catches the cross town bus to find him. </p>
  <h2><a name="climax">Climax</a></h2>
  <p>when Frankie gets to Nellie's house, she looks up and sees them
kissing on the balcony. With tears in her eyes, she picks up her
shotgun and kills her Johnny. He falls to the ground. </p>
  <h2 id="denoue"> Denouement</h2>
  <p> Frankie feels bad but it's kind of late now, and Johnny
<em>was</em> a lech. But the police come and cart her off
anyway.</p></body>

```



شکل ۵-۳



شکل ۵-۴

اگر مقصد لنگر در یک صفحه مجزا باشد، برای اشاره کردن به یک بخش از `` استفاده شود. (هیچ فضای خالی نباید بین URL و # وجود داشته باشد). اگر مقصد لنگر روی یک صفحه روی یک سرور متفاوت قرار داشته باشد می‌بایست از دستور `` استفاده کرد (بدون هیچ فضای خالی).

مقصد لنگر می‌تواند هر عنصری غیر از تگ a نیز باشد. در این حالت می‌بایست از صفت id در این عنصر با یک مقدار مجزا در کل صفحه استفاده شود.

لینک های دانلود

لینک های دانلود به لینک هایی گفته می‌شود که مسیر دسترسی به یک فایل خاص بر روی سرور خودمان یا سایر سرور ها را مشخص می‌کند. معمولا از این نوع لینک ها برای فراهم کردن امکان دانلود فایل ها به کاربران استفاده می‌شود. در صورتی که بر روی لینک های دانلود کلیک کنید چنانچه مرورگر خود نتواند فایل مورد نظر را باز کند شروع به دانلود آن خواهد کرد. در مثال زیر نمونه ساده یک لینک دانلود که امکان دانلود فایلی به نام html_learning.zip را فراهم می‌کند مشاهده می‌کنید:

```
<a href="itcom.pnuab.ac.ir/ebooks/html_learning.zip" >Click to  
Download</a>
```


فصل ششم

کار با تصاویر

امروزه دیگر نقش تصاویر در جذابیت هر چه بیشتر صفحات وب و خارج کردن آن ها از حالت یکنواختی غیر قابل انکار است. ما در این فصل نحوه گنجاندن انواع تصاویر را خواهیم آموخت و با روش تبدیل عکس ها به لینک آشنا خواهیم شد.

تگ

برای قرار دادن عکس ها از تگ که جزء تگ های تهی است استفاده می شود:

```

```

کد ۶-۱

صفت های این تگ به شرح زیر است:

align: این صفت نحوه قرار گیری عکس نسبت به متون و عناصر اطرافش در صفحه را کنترل می کند که می تواند مقادیر زیر را بپذیرد:

right: عکس را در سمت راست متون اطرافش قرار می دهد.

left: عکس را در سمت چپ متون اطرافش قرار می دهد.

top: قسمت بالای عکس با خطی که در آن قرار دارد تراز خواهد شد.

middle: میانه های عکس با خطی که در آن قرار دارد تراز خواهد شد.

bottom: قسمت پایینی عکس با خطی که در آن قرار دارد تراز خواهد شد.

alt: متنی را به عنوان مقدار می پذیرد و در صورتی که به هر دلیلی عکس مورد نظر پیدا نشود به جای عکس نمایش داده خواهد شد. در نسخه XHTML شما باید حتما از صفت alt برای عکس هایتان استفاده کنید. حتی خالی !!! موتور های جستجو از مقدار تعیین شده برای این صفت به منظور بدست آوردن اطلاعاتی در مورد عکس ها استفاده می کنند.

border: میزان ضخامت حاشیه عکس را با عددی بر حسب px مشخص می کند.

height: ارتفاع عکس را بر حسب px یا درصد بیان می کند.

hspace: میزان فاصله سمت چپ و راست عکس نسبت به عناصر اطراف را کنترل می کند.

src: این صفت اصلی ترین صفت برای تگ img است و آدرس عکسی که می خواهیم نمایش داده شود را مشخص می کند. مقدار این صفت می تواند یکی از انواع آدرس های نسبی و مطلق باشد.

vspace: میزان فاصله سمت بالا و پایین عکس نسبت به عناصر اطراف را کنترل می کند.

width: عرض عکس را بر حسب px یا درصد بیان می کند.

تصاویر به عنوان لینک

تبدیل تصاویر به لینک بسیار آسان است. بدین منظور کافی است به جای قرار دادن یک متن ساده در بین تگ های باز و بسته a از یک تگ img که اشاره به عکس مورد نظران دارد استفاده کنید. تصاویر در این حالت معمولاً برای ایجاد دکمه های گرافیکی یا لینک به سایر صفحات همانند مثال زیر استفاده می شوند:

```
<a href="../../index.html" title="Click here to return to the home page">
  
</a>
```

کد ۶-۲

درج توضیحات در HTML

همانند بسیاری از زبان های برنامه نویسی امکان گنجانیدن توضیحات در صفحات HTML نیز وجود دارد. معمولاً از توضیحات در HTML برای خواناتر کردن کد ها و مشخص کردن نقاط ابتدایی و انتهایی بخش های صفحه استفاده می شود. یک توضیح در HTML به شکل زیر است:

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

کد ۶-۳

همانطور که می بینید یک comment با <! شروع و به > ختم می شود. در حقیقت متنی که بین دو -- ابتدایی و دو -- انتهایی قرار می گیرد توسط مرورگر ها به عنوان comment تعبیر خواهد شد. دقت داشته باشید که بین دو -- ابتدایی و انتهایی هیچ -- نباید قرار بگیرد. به عنوان مثال دستور زیر نادرست است:

```
<p>Take the next right.<!-- Look out for -- Castle --></p>
```

کد ۶-۴

یک دستور comment می تواند حاوی بیش از یک متن به عنوان توضیحات باشد. به عنوان مثال دستور زیر دو comment را تعریف می کند:

```
<p>Take the next right.<!-- Look out for -- -- Castle --></p>
```

کد ۶-۵

در دستور فوق عبارت Look out for یک comment و عبارت Castle یک comment دیگر است.

دقت داشته که تعداد کاراکترهای - در یک comment صحیح باید همیشه مضربی از ۴ باشد!

می‌توان از comment ها برای از کار انداختن تگ های یک صفحه استفاده کرد. به عنوان مثال در کد زیر دیگر محتوای تگ h2 در صفحه نمایش داده نخواهد شد.

```
<!-- <h2>This Paragraph Not Display in Page</h2> -->
```

کد ۶-۶

اما نمی‌توان از comment ها برای از کار انداختن صفت های داخل یک تگ استفاده کرد. به عنوان مثال کد زیر نمی‌تواند از عملی که صفت title انجام می‌دهد جلوگیری کند:

```
<p <!--title="My Paragraph" --> >This is a Paragraph</p>
```

کد ۷-۶

فصل هفتم

کار با لیست ها

لیست ها یکی از پرکاربردترین عناصر موجود در HTML می باشند که طراحان را قادر به نمایش انواع اطلاعات به صورت لیست می نماید. در HTML شما می توانید سه نوع لیست ایجاد کنید: لیست های مرتب، لیست های نامرتب و لیست های ترتیبی. همچنین امکان ترکیب انواع لیست ها با یکدیگر به منظور ایجاد لیست های تو در تو¹ نیز فراهم شده است. امروزه از لیست ها به همراه زبان CSS برای کنترل ظاهر نمایشی آن ها، به منظور ایجاد جلوه های گرافیکی همچون منو های درختی، منوهای افقی و عمودی، tab ها و... استفاده می شود.

¹ Nested List

دسته بندی لیست ها

لیست ها به سه دسته اصلی زیر تقسیم بندی می شوند:

لیست های نامرتب^۱

در این نوع لیست ها ترتیب قرار گیری عناصر لیست (که غالبا متون ساده) هستند مهم نبوده و معمولا برای لیست کردن آن ها از bullet های دایره ای توپر، ساده یا مربع استفاده می شود.

- Eggs
- Cheese
- Milk
- Papadums
- Tickle-me Elmo
- Dr Who Remote Control Dalek

لیست های مرتب^۲

در این نوع لیست ها ترتیب قرار گیری عناصر لیست مهم بوده و برای نمایش ترتیب آن ها از اعداد یا حروف انگلیسی یا رومی استفاده می شود.

1. Remove the outer casing by pushing the plastic rivets through.
2. Disconnect the main power harness from the inner unit (unclip).
3. Remove connection to the glow plug.
4. Extract unit, keeping upright at all times

لیست های تعریفی^۳

از این گونه لیست ها هنگامی که می خواهیم یکسری از آیتم هایی که یک بخش عنوان و یک توضیح برای آن عنوان دارند را نمایش دهیم استفاده می کنیم.

Spam

Unsolicited email sent in the hope of increasing sales of some product, or simply for the purposes of annoying people

Spammer

Someone who sends out spam email and therefore deserves to develop a nasty incurable disease of some kind

Spam Filter

a tool used in email to 'filter out' likely spam messages, usually placing them in a dedicated junk messages folder or similar

در ادامه با تگ های مربوط به هر یک از انواع لیست ها آشنا خواهیم شد:

¹ unordered list

² ordered list

³ Definition List

تگ برای ایجاد لیست های مرتب

از این تگ برای ایجاد لیست های مرتب استفاده می شود. برای ایجاد آیتم های لیست ها می بایست از تگ (مخفف list Item) در داخل تگ استفاده نمود. این تگ از صفت های زیر پشتیبانی می کند:

Type: با استفاده از این صفت می توان نوع bullet کنار هر آیتم را مشخص نمود. این صفت مقادیر زیر را می پذیرد:

a: برای نمایش حروف کوچک انگلیسی

A: برای نمایش حروف بزرگ انگلیسی

i: برای نمایش حروف کوچک رومی

I: برای نمایش حروف بزرگ رومی

1: برای نمایش اعداد به عنوان bullet برای آیتم ها (مقدار پیش فرض)

Start: این صفت مشخص می کند که شماره گذاری لیست ما باید از چندمین کاراکتر از انواعی که در صفت type مشخص شده است شروع شود. به عنوان مثال اگر مقدار صفت type='A' مقدار 3 برای صفت start به معنی کاراکتر C خواهد بود.

Compact: این صفت بولین موجب فشردن نمایش داده شدن آیتم های لیست خواهد شد. البته این صفت در هیچ مرورگری پشتیبانی نمی شود.

تگ برای ایجاد لیست های نامرتب

از این تگ برای ایجاد لیست های نامرتب استفاده می شود. برای ایجاد آیتم های لیست ها می بایست از تگ در داخل تگ استفاده نمود. این تگ از صفت های زیر پشتیبانی می کند:

Type: با استفاده از این صفت می توان نوع bullet کنار هر آیتم را مشخص نمود. این صفت مقادیر زیر را می پذیرد:

circle: یک دایره توخالی را به عنوان bullet نمایش می دهد

Disc: یک دایره توپر (سیاه رنگ) را به عنوان bullet نمایش می دهد

Square: یک مربع توپر را به عنوان bullet نمایش می دهد.

Compact: این صفت بولین موجب فشردن نمایش داده شدن آیتم های لیست خواهد شد. البته این صفت در هیچ مرورگری پشتیبانی نمی شود.

تگ <dl> برای ایجاد لیست های تعریفی

برای ایجاد لیست های تعریفی از تگ <dl> استفاده میشود. هر لیست تعریفی از چند عنوان و توضیحات آن عنوان تشکیل می‌شود. برای ایجاد هر عنوان از تگ <dt>^۲ و برای ایجاد هر توضیح از تگ <dd>^۳ استفاده می‌شود. تگ <dl> فقط از صفت compact پشتیبانی می‌کند. در زیر شما می‌توانید یک لیست تعریفی را مشاهده نمایید :

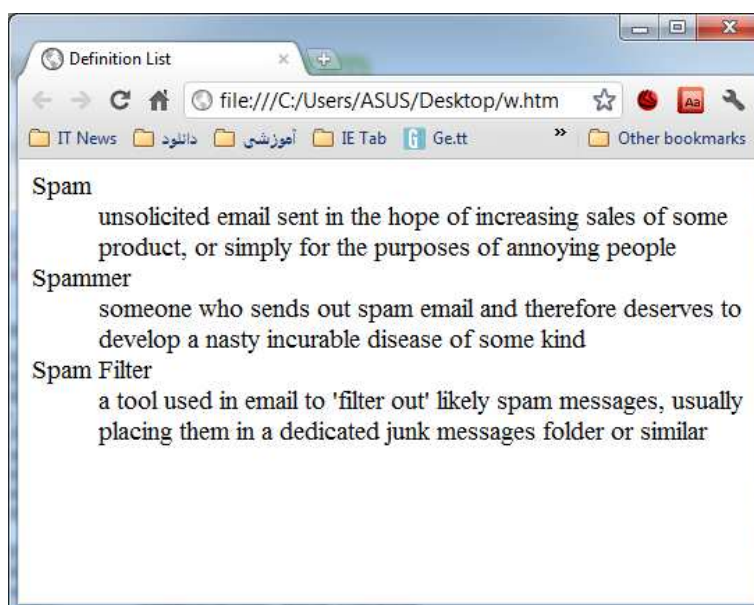
```
<dl>
  <dt>Spam</dt>
  <dd>unsolicited email sent in the hope of increasing sales of
  some product, or simply for the purposes of annoying people
  </dd>

  <dt>Spammer</dt>
  <dd>someone who sends out spam email and therefore deserves to
  develop a nasty incurable disease of some kind
  </dd>

  <dt>Spam Filter</dt>
  <dd>a tool used in email to 'filter out' likely spam messages,
  usually placing them in a dedicated junk messages folder
  or similar
  </dd>
</dl>
```

کد ۱-۷

حاصل اجرای کد فوق در مرورگر به صورت زیر خواهد بود:



شکل ۱-۷

¹ Definition List

² Definition title(term)

³ Definition description

تگ

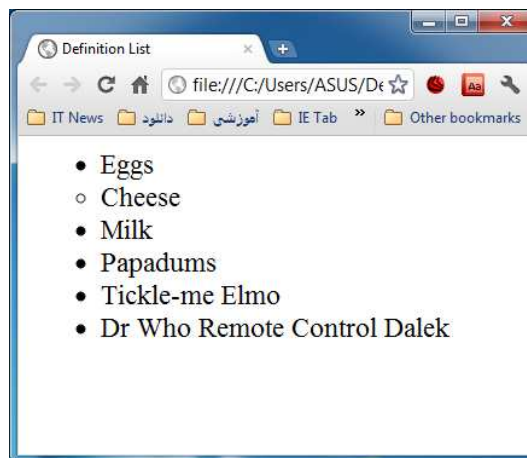
از این عنصر برای مشخص کردن آیتم های لیست های مرتب و نامرتب استفاده می‌شود. غالباً این تگ داخل تگ های و استفاده می‌شود. این تگ از صفت های زیر پشتیبانی می‌کند:

type: از این صفت برای تغییر دادن نوع bullet مربوط به یک آیتم لیست به غیر از آن نوعی که در صفت type تگ های ul و ol تنظیم شده است استفاده می‌شود. این صفت برای تگ همان مقادیری را که در تگ می‌گیرد به عنوان مقدار می‌پذیرد. به مثال زیر و حاصل آن در مرورگر دقت کنید:

```
<ul type='disc'>
  <li>Eggs</li>
  <li type="circle">Cheese</li>
  <li>Milk</li>
  <li>Papadums</li>
  <li>Tickle-me Elmo</li>
  <li>Dr Who Remote Control Dalek</li>
</ul>
```

کد ۷-۲

حاصل به صورت زیر خواهد بود:



شکل ۷-۲

value: از این صفت برای تغییر دادن عدد کنار یک آیتم از لیست استفاده می‌شود. در واقع ما می‌توانیم ترتیب نمایش اعداد کنار یک آیتم را با این صفت به هم بزنییم. این صفت یک عدد صحیح را به عنوان مقدار می‌پذیرد. به مثال زیر دقت کنید:

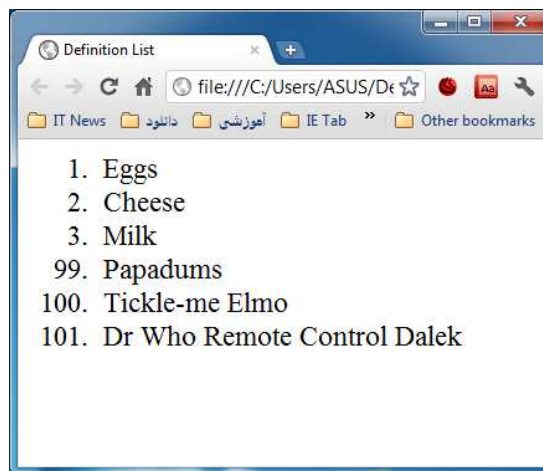
```

<ol>
  <li>Eggs</li>
  <li>Cheese</li>
  <li>Milk</li>
  <li value="99">Papadums</li>
  <li>Tickle-me Elmo</li>
  <li>Dr Who Remote Control Dalek</li>
</ol>

```

کد ۳-۷

حاصل اجرای کد فوق به شکل زیر خواهد بود:



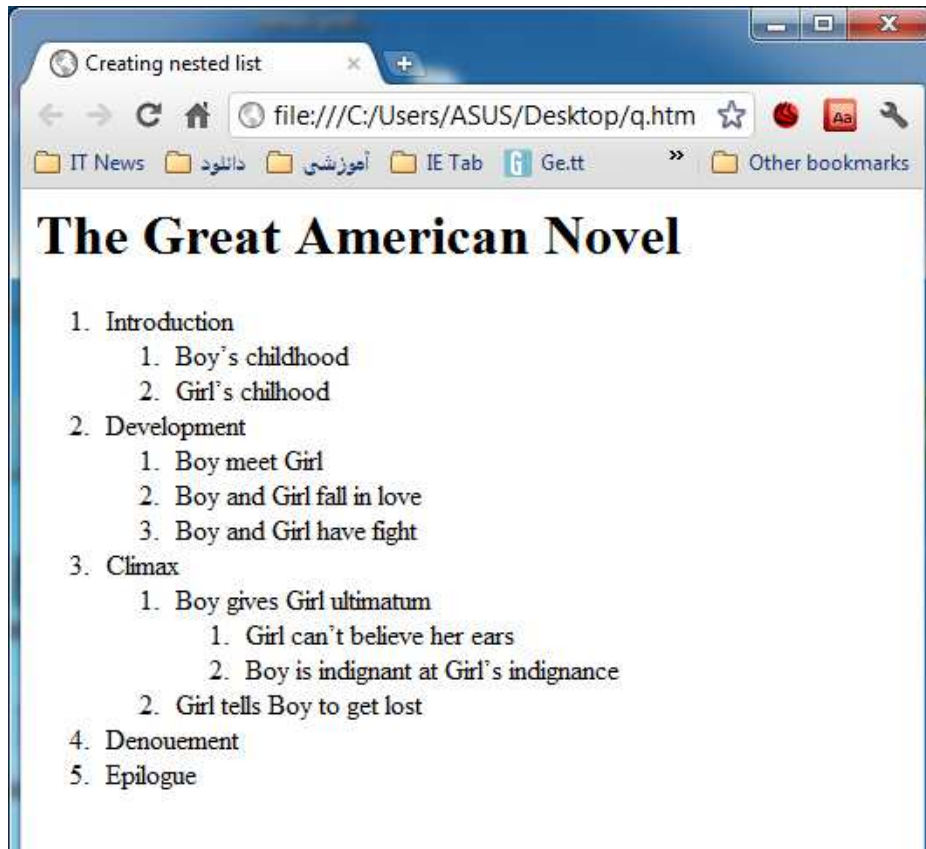
شکل ۳-۷

لیست های تو در تو^۱

یکی از ویژگی های بارز لیست ها امکان ترکیب کردن آن ها با یکدیگر است. امروزه ترکیب لیستها کاربرد های فراوانی پیدا کرده است و در بسیاری از موارد همچون ایجاد ساختار های درختی، منوهای افقی و عمودی چندسطحی و... استفاده می شود. برای ایجاد لیست های تو در تو کافی است هر لیست جدید را در داخل یک عنصر `li` قرار دهید. در ادامه کد یک نمونه لیست تودرتو به همراه حاصل آن در مرورگر را مشاهده می کنید:

¹ nested Lists

```
<h1>The Great American Novel</h1>
<ol>
  <li> Introduction
    <ol>
      <li>Boy's childhood</li>
      <li>Girl's childhood</li>
    </ol>
  </li>
  <li>Development
    <ol>
      <li>Boy meet Girl</li>
      <li>Boy and Girl fall in love</li>
      <li> Boy and Girl have fight</li>
    </ol>
  </li>
  <li>Climax
    <ol>
      <li>Boy gives Girl ultimatum
        <ol>
          <li> Girl can't believe her ears</li>
          <li>Boy is indignant at Girl's indignance</li>
        </ol>
      </li>
      <li>Girl tells Boy to get lost</li>
    </ol>
  </li>
  <li>Denouement </li>
  <li>Epilogue </li>
</ol>
```



شکل ۷-۴

فصل هشتم

کار با جداول

جداول یکی از مهمترین عناصر موجود در HTML به شمار می‌روند و معمولاً برای نظم دهی به سایر عناصر موجود در صفحه همچون فرم‌ها، لینک‌ها، تصاویر، متن‌ها و... استفاده می‌شوند. در ابتدا جداول به منظور قالب بندی و طرح بندی کلی صفحات کاربرد ویژه‌ای داشتند اما امروزه با ظهور CSS و پیدایش روش‌های نوین طرح بندی کاربردهای آنها به نمایش منظم سایر عناصر محدود شده است.

در این فصل با نحوه استفاده از جداول و تگ‌ها و صفت‌های مربوط به آنها آشنا خواهیم کرد و در ادامه ویژگی‌های پیشرفته‌تر آنها یعنی گروه بندی‌های ردیفی و ستونی را بررسی خواهیم کرد.

ساختار اساسی جدول

ساختار کلی یک جدول در مثال زیر آمده است:

```
<table border="1">
  <tr>
    <th>Account Type</th>
    <th>Interest Rate</th>
  </tr>
  <tr>
    <td>Smart</td>
    <td>From 2%</td>
  </tr>
  <tr>
    <td>Young Saver</td>
    <td>From 1.6%</td>
  </tr>
</table>
```

کد ۸-۱

حاصل اجرای کد فوق در مرورگر به شکل زیر خواهد بود:

Account Type	Interest Rate
Smart	From 2%
Young Saver	From 1.6%

شکل ۸-۱

در حالت کلی می‌توان گفت که جدول‌ها در HTML از ردیف‌هایی که سلول‌ها در آن‌ها قرار دارند تشکیل می‌شوند. در ادامه با تگ‌های مرتبط با ساختار جداول بیشتر آشنا خواهیم شد:

تگ <table>



برای ایجاد یک جدول استفاده می‌شود. در واقع ابتدا و انتهای هر جدول را مشخص می‌کند. تگ <table> از صفت‌های زیر پشتیبانی می‌کند:

border: میزان ضخامت حاشیه اطراف جدول را مشخص می‌کند.



dir: جهت‌گیری متون داخل جدول را مشخص می‌کند و می‌تواند یکی از مقادیر rtl یا ltr را اختیار می‌کند.



bordercolordark: معمولا border اطراف جدول از دو رنگ تیره و روشن ایجاد می‌شود این صفت رنگ تیره تر حاشیه را تعیین می‌کند.

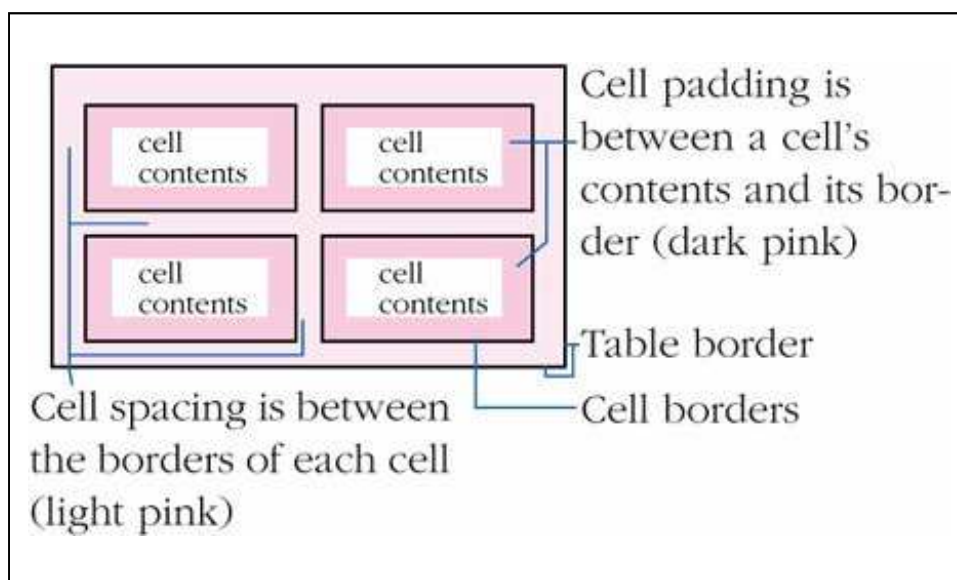
bordercolorlight: رنگ روشن border را مشخص می‌کند.

bordercolor: رنگ اصلی border را تعیین می‌کند و آن را از حالت دو رنگی خارج می‌کند. در واقع border را فقط به یک رنگ نمایش می‌دهد.

cellpadding: عددی صحیح بر حسب px را به عنوان مقدار می‌پذیرد و فضای محتویات داخل سلولها و دیواره هایشان را کنترل می‌کند. (البته بر حسب درصد هم می‌تواند باشد که در این صورت نسبت به عرض کل سلول سنجیده خواهد شد.)

cellspacing: عددی صحیح بر حسب px را به عنوان مقدار می‌پذیرد و فضای بین هر سلول را کنترل می‌کند.

تصویر زیر موقعیت صفت های border، cellpadding و cellspacing را به شما نمایش می‌دهد:



شکل ۸-۲

align: محل قرار گیری جدول در صفحه (یا در عنصری که جدول در آن قرار دارد) را تعیین می‌کند و یکی از مقادیر زیر را می‌پذیرد:

Right: جدول در سمت راست قرار می‌گیرد

Center: جدول در وسط قرار می‌گیرد

Left: جدول در سمت چپ قرار می‌گیرد

width: عرض جدول را بر حسب px یا درصد مشخص می‌کند.

height: ارتفاع جدول را بر حسب px یا درصد مشخص می‌کند.

bgcolor: رنگی را برای پس زمینه جدول مشخص می‌کند.

summary: رشته ای ساده که شامل توضیحاتی در مورد محتویات جدول است را به عنوان مقدار می‌پذیرد.

background: آدرس عکسی را به عنوان مقدار می‌پذیرد و آن را در پس زمینه جدول قرار می‌دهد. در صورتی که عکس از جدول کوچکتر باشد آن را هم در جهت افقی و هم در جهت عمودی در پس زمینه تکرار خواهد کرد.

frame: مشخص می‌کند که کدام یک از border های چهار طرف جدول نمایش داده شود. این صفت مقادیر زیر را می‌پذیرد:

void: هیچ کدام از border ها را نمایش نمی‌دهد. (مقدار پیش فرض)

hsides: فقط border های افقی را نمایش می‌دهد.

vsides: فقط border های عمودی را نمایش می‌دهد.

rhs: فقط border سمت راست را نمایش می‌دهد.

lhs: فقط border سمت چپ را نمایش می‌دهد.

above: فقط border بالا را نمایش می‌دهد.

below: فقط border پایینی را نمایش می‌دهد.

border یا **box**: همه border ها را نمایش می‌دهد.

rules: مشخص می‌کند که کدام یک از حاشیه های داخلی جدول (اطراف ردیف، گروهی از ردیف ها، سلول ها و گروهی از سلول ها) نمایش داده شوند. این صفت مقادیر زیر را می‌پذیرد:

none: هیچ کدام از border های داخلی را نمایش نمی‌دهد (مقدار پیش فرض)

all: تمام border های داخلی را نمایش می‌دهد.

groups: فقط حاشیه های گروهی از ردیف ها یا گروهی از ستون ها را نمایش می‌دهد.

cols: فقط border ستون ها را نمایش می‌دهد.

rows: فقط border ردیف ها را نمایش می‌دهد.

تگ **<tr>**

برای ایجاد یک ردیف و همیشه در داخل تگ **<table>** استفاده می‌شود. هر جدول می‌بایست حداقل یک ردیف داشته باشد. این تگ صفت های زیر را داراست:

align: چگونگی چینش متن در هر یک از سلول های آن ردیف را مشخص می کند. و می تواند یکی از مقادیر زیر را بپذیرد:

- Left
- right
- center
- justify

bgcolor: رنگ پس زمینه ردیف را مشخص می کند.

valign: تراز عمودی متن در هر یک از سلول های یک ردیف را مشخص می کند و مقادیر زیر را می پذیرد:

- top**: محتوای سلول را با بالای سلول تراز می کند.
- middle**: محتوای سلول را با وسط سلول تراز می کند.
- bottom**: محتوای سلول را با پایین سلول تراز می کند.
- baseline**: محتوای سلول را با اولین خطی که محتوای سلول ها از آن آغاز می شود همتراز می کند.

تگ های <th> و <td>

از این دو تگ برای ایجاد سلول های داخل هر ردیف استفاده می شود. تگ <td> برای سلول هایی که شامل داده های جدولی ساده (table data) و تگ <th> برای سلول هایی که شامل عنوان جدولی (table heading) هستند استفاده می شود. به صورت پیش فرض محتوای سلول های از نوع heading به صورت bold و وسط چین و سلول های ساده به صورت نرمال و چپ چین نمایش داده می شوند. این دو تگ صفت های مشابهی دارند که در زیر بیان می شود:

abbr: این صفت مخففی از محتوای سلول را نشان می دهد. به مثال زیر توجه کنید:

```
<td abbr='pnu'>Payam Nour University</td>
```

کد ۸-۲

align: نحوه چینش محتوای سلول را در جهت افقی کنترل می کند و یکی از مقادیر Left, Right, Center و Justify را اختیار می کند.

valign: تراز عمودی محتوای سلول را کنترل می کند و می تواند یکی از مقادیر top, middle, bottom, baseline را اختیار کند.

bgcolor: رنگ پس زمینه سلول را تعیین می کند.

مقدار bgcolor در یک سلول (th یا td) بر رنگ مشخص شده در یک سطر (tr) و رنگ مشخص شده برای یک سطر به نوبه خود رنگ مشخص شده برای گروهی از سطر ها یا ستون ها (در تگ thead، colgroup و ...) و رنگ مشخص شده برای گروهی از سطر ها و ستون ها هم بر رنگ مشخص شده کل جدول غلبه می‌کند.

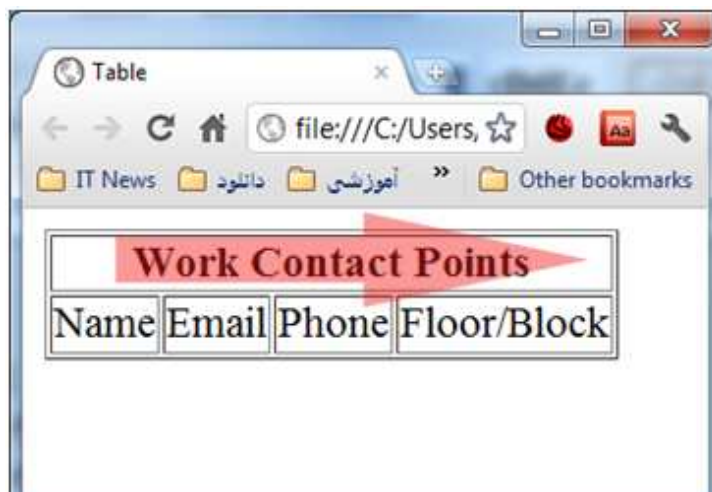
width: عرض سلول را مشخص می‌کند

height: ارتفاع سلول را مشخص می‌کند.

Nowrap: این صفت بولین از شکسته شدن متن موجود در یک سلول جلوگیری می‌کند مگر اینکه در متن داخل سلول از تگ `
` استفاده شده باشد.

Colspan: از این صفت هنگامی که می‌خواهیم یک سلول را در طول چندین ستون گسترش بدهیم استفاده می‌کنیم. عددی صحیح را به عنوان مقدار می‌پذیرد که این عدد مشخص کننده ی تعداد ستون هایی است که باید سلول در آن ها گسترش پیدا کند. به مثال زیر و حاصل آن در مرور گر توجه کنید:

```
<table border="1">
  <tr>
    <th colspan="4">Work Contact Points</th>
  </tr>
  <tr>
    <td>Name</td>
    <td>Email</td>
    <td>Phone</td>
    <td>Floor/Block</td>
  </tr>
  .
  .
  .
</table>
```



شکل ۸-۳

Rowspan: از این صفت هنگامی که می‌خواهیم یک سلول را در طول چندین ردیف گسترش دهیم استفاده می‌کنیم. عددی صحیح را به عنوان مقدار می‌پذیرد که این عدد مشخص کننده ی تعداد ردیف هایی است که باید سلول در آن ها گسترش پیدا کند. به مثال زیر و حاصل آن در مرور گر توجه کنید:

```
<table border="1">
  <tr>
    <td>Bill</td>
    <td>bill@megacorp.com</td>
    <td>345678</td>
    <th rowspan="3">3/C</th>
  </tr>
  <tr>
    <td>Jane</td>
    <td>jane@megacorp.com</td>
    <td>777444</td>
    ***missing cell taken up by th with rowspan***
  </tr>
  <tr>
    <td>Alison</td>
    <td>alison@megacorp</td>
    <td>888652</td>
    ***missing cell taken up by th with rowspan***
  </tr>
</table>
```

کد ۸-۴

حاصل در مرورگر:

Bill	bill@megacorp.com	345678	3/C
Jane	jane@megacorp.com	777444	
Alison	alison@megacorp	888652	

شکل ۸-۴

تگ <caption>

از این تگ برای ایجاد یک عنوان برای جدول استفاده می‌شود. این تگ باید بلافاصله بعد از تگ آغازین table و قبل از اولین ردیف جدول قرار گیرد. به صورت زیر:

```
<table>
  <caption>the table caption come here</caption>
<tr>
  .
  .
  .
```

کد ۸-۵

به صورت پیش فرض مرورگرها متن داخل این تگ را در بالا و وسط جدول نمایش می‌دهند. تگ <caption> از صفت زیر پشتیبانی می‌کند:

align: محل قرارگیری عنوان جدول را مشخص می‌کند و مقادیر زیر را می‌پذیرد:

- right**: عنوان در سمت راست جدول نمایش داده می‌شود
- left**: عنوان جدول در سمت چپ جدول نمایش داده می‌شود
- top**: عنوان جدول در بالای جدول نمایش داده می‌شود.
- bottom**: عنوان جدول در پایین جدول نمایش داده می‌شود.

گروه بندی ردیفی و ستونی جداول

ما می‌توانیم سلول‌های جدول را به گروهی از ستون‌ها و گروهی از ردیف‌ها تقسیم بندی کرده و خواص مشابهی را به آن‌ها اعمال کنیم. ابتدا به گروه بندی‌های ردیفی می‌پردازیم:

ما می‌توانیم ردیف‌های جدول را به سه نوع گروه ردیفی تقسیم بندی کنیم: **head**, **body**, **foot**. برای این کار از سه تگ **<thead>**, **<tbody>**, **<tfoot>** استفاده می‌کنیم.

هر جدول می‌تواند شامل چندین عنصر `<tbody>` اما فقط دارای یک عنصر `<tfoot>` و `<thead>` باشد. نکته دیگر اینکه تگ `<tfoot>` همیشه باید قبل از `<tbody>` و بعد از `<thead>` قرار بگیرد.

هر سه این عنصرها دارای صفت های `align` و `valign` می‌باشند که قبلا به آن‌ها اشاره شده است. معمولا از گروه بندی های ردیفی هنگامی استفاده می‌شود که جدول بسیار طولانی است. یکی از مزیت های این نوع گروه بندی هنگام `print` گرفتن از این نوع جدول هاست که معمولا چند صفحه ای می‌شوند و می‌خواهیم که عنوان و پایه جداول در همه صفحه های چاپی تکرار شوند. یک نمونه از گروه بندی ردیفی را در زیر مشاهده می‌کنید.

```
<table>
  <thead>
    <tr><td>.....</td></tr>
    <tr><td>.....</td></tr>
  </thead>

  <tfoot>
    <tr><td>.....</td></tr>
    <tr><td>.....</td></tr>
  </tfoot>

  <tbody>
    <tr><td>.....</td></tr>
    <tr><td>.....</td></tr>
  </tbody>

  <tbody>
    <tr><td>.....</td></tr>
    <tr><td>.....</td></tr>
  </tbody>
</table>
```

کد ۸-۶

به ساختار جدول فوق و ترتیب ظاهر شدن عناصر `thead` , `tfoot` , `tbody` که دقیقا به همین ترتیب می‌بایست ظاهر شوند دقت کنید.

همیشه ردیف هایی که داخل تگ `<thead>` قرار می‌گیرند در بالای جدول و ردیف هایی که در داخل تگ `<tfoot>` قرار می‌گیرند در پایین جدول ظاهر می‌شوند.

گروه بندی های ستونی

به وسیله این نوع گروه بندی می‌توانیم سلول های چندین ستون را به نوعی با هم مرتبط کرده و یکسری خواص مشترک را برای آن‌ها تعیین کنیم. برای گروه بندی های ستونی به دو طریق می‌توان عمل کرد:

استفاده از تگ <col />

به وسیله این تگ تهی که باید بلافاصله بعد از تگ آغازین جدول قرار بگیرد می‌توان یک یا چند ستون را به هم مربوط و در واقع با هم گروه بندی کرد:

بدین منظور هر عنصر <col /> نشان دهنده یک گروه ستونی خواهد بود. اما می‌توان با استفاده از صفت span مشخص کرد که گروه ما شامل چند ستون از جدول باشد. به عنوان مثال در کد زیر سه گروه تشکیل می‌شود که گروه ستونی اول شامل اولین ستون جدول، گروه دوم شامل سه ستون بعدی آن و گروه سوم شامل دو ستون بعد از آن ها خواهد بود:

```
<table>
  <col align='right' span='1' />
  <col align='left' span='3' />
  <col align='center' span='2' />
  .
  .
  .
```

کد ۸-۷

استفاده از تگ <colgroup>

یکی دیگر از راه های ایجاد گروه های ستونی استفاده از تگ <colgroup> است. کاربرد این تگ برای ایجاد گروه های ستونی همانند کاربرد تگ <col> می‌باشد. به این معنی که هر تگ <colgroup> به صورت پیش فرض شامل یک ستون از جدول خواهد بود اما می‌توانیم با استفاده از صفتی به نام span یک گروه ستونی شامل چندین ستون را ایجاد کنیم. به عنوان مثال کد زیر همان نتیجه ای را خواهد داشت که کد فوق با استفاده از تگ <col> داشت:

```
<table>
  <colgroup align='right' span='1'></colgroup>
  <colgroup align='left' span='3'></colgroup>
  <colgroup align='center' span='2'></colgroup>
  .
  .
  .
```

کد ۸-۸

دقت کنید که تگ colgroup جزء تگ های تهی نیست اما محتوای آن در این حالت می‌بایست خالی باشد. برای مشخص کردن اینکه یک گروه ایجاد شده بوسیله تگ <colgroup> شامل چند ستون از جدول باشد دو راه وجود دارد: یکی استفاده از صفت span که تعداد گروه های هر ستون را مشخص می‌کند. به این مورد در بالا اشاره کردیم. راه دیگر استفاده از تعدادی تگ <col> داخل تگ <colgroup> است در واقع تعداد تگ های <col> به کار رفته در داخل هر <colgroup> تعداد ستون های آن گروه ستونی را مشخص می‌کند. به عنوان مثال کد زیر دقیقاً همان نتیجه ای را خواهد داشت که کد فوق با استفاده از صفت span می‌کرد:

```

<table>
  <colgroup align='right'>
    <col />
  </colgroup>

  <colgroup align='left'>
    <col />
    <col />
    <col />
  </colgroup>

  <colgroup align='center'>
    <col />
    <col />
  </colgroup>
  .
  .
  .

```

کد ۸-۹

اصطلاحاً به گروه های ستونی که با استفاده از تگ **<colgroup>** ایجاد می شوند گروه های ستونی **ساختاری** و به گروه هایی که با استفاده از تگ **<col>** ایجاد می شوند گروه های ستونی **غیر ساختاری** می گویند. تفاوت گروه های ساختاری و غیر ساختاری هنگامی است که از صفت `rules='groups'` برای نمایش مرز گروه ها در جداول استفاده می شود است. در این صورت فقط مرز گروه های ستونی ساختاری نمایش داده خواهد شد.

فصل نهم

کار با فرم ها

یکی دیگر از مهمترین عناصر موجود در HTML فرم ها هستند. عناصر فرم تنها عناصری هستند که کاربران می توانند مستقیما یکسری اطلاعات را در آن ها وارد کرده و برای طراح (سازنده) بفرستند. از جمله عناصر فرم می توان به کادر های متنی یک خطی، کادر های متنی چند خطی، دکمه ها، دکمه های رادیویی، دکمه های تصویری، کادر های علامت، لیست های انتخاب و... اشاره کرد..

در این فصل ابتدا به سراغ ساختار اساسی فرم ها و سپس به بررسی دقیق هر یک از عناصر فوق الذکر با ذکر جزئیات و کاربردهایشان خواهیم پرداخت.

ساختار اساسی فرم

هر فرم با تگ `<form>` شروع و به `</form>` نیز ختم می‌شود. تگ فرم می‌تواند شامل سایر تگ های HTML همچون پاراگراف ها، تیترا ها و مانند آن ها باشد اما نمی‌توان از تگ `<form>` دیگری در داخل آن استفاده کرد. هر صفحه می‌تواند شامل چندین فرم که هر یک وظایف مجزایی از یکدیگر دارند باشد. به عنوان مثال شما ممکن یک فرم ورود، یک فرم جستجو و فرمی برای مشترک کردن کاربرانتان در خبرنامه سایتتان داشته باشید.

یک فرم دارای سه بخش مهم می‌باشد: تگ `form`، که شامل URL صفحه ای که اطلاعات فرم را پردازش می‌کند، عناصر فرم مانند کادر های متنی، کادر های علامت، لیست ها و یک دکمه `submit` که داده ها را برای صفحه پردازشگر رویداد سرور می‌فرستد. مثال زیر ساختار اصلی یک فرم را نشان می‌دهد:

```
<form method="post" action="showform.php">
  <p class="legend">personal information </p>
  <fieldset id="personal">
    <label>Name:</label>
    <input type="text" name="name" size="30"/><br/>
    <label>Address:</label>
    <input type="text" name="address" size="30"/><br/>
    <label>Town/City:</label>
    <input type="text" name="city" size="30"/><br/>
    ...
  <p id="buttons">
    <input type="submit" value="Order Bed"/>
    <input type="reset" value="Start Over"/>
  </p>
</form>
```

کد ۹-۱

حاصل اجرای کد فوق در مرورگر به صورت زیر خواهد بود:

The screenshot shows a web browser window with the title 'Form'. The address bar shows a local file path: 'file:///C:/Users/ASUS/Desktop/w.h'. The browser's bookmark bar includes 'IT News', 'دانلود', 'آموزشی', 'IE Tab', 'Ge.tt', and 'Other bookmarks'. The main content area displays the rendered HTML form from the code block above, titled 'personal information'. It features three text input fields for 'Name', 'Address', and 'Town/City', followed by two buttons: 'Order Bed' and 'Start Over'.

شکل ۹-۱

در ادامه با هر یک از عناصر فرم و نحوه ایجاد آن ها آشنا خواهیم شد:

تگ <form>

این تگ برای ایجاد فرم‌ها استفاده می‌شود. در واقع هر فرمی به وسیله تگ باز <form> شروع می‌شود و با تگ بسته </form> پایان می‌یابد. عناصر فرم باید حتماً داخل این عنصر به کار روند. این تگ شامل تعدادی صفت است که در ادامه بررسی خواهیم کرد:

action: فرم‌ها بعد از اینکه توسط کاربر پر می‌شوند می‌بایست برای یک برنامه پردازشگر که معمولاً به یکی از زبان‌های تحت سرور^۱ همانند php، asp.net، jsp و... نوشته می‌شوند فرستاده شوند. ما باید در این صفت آدرس صفحه پردازشگر که قرار است فرم به آن ارسال شود را مشخص کنیم.

method: همانطور که از اسم این صفت پیداست روش ارسال فرم را مشخص می‌کند. یک فرم می‌تواند به دو روش به صفحه پردازشگر ارسال شود: post و get. این روش‌ها تفاوت‌هایی با هم دارند که تعدادی از آن‌ها را در ادامه بررسی می‌کنیم:

متد get از URL (آدرس) صفحه برای ارسال داده‌ها استفاده می‌کند و آن‌ها را در قالب جفت‌های name=value به آدرس صفحه پردازشگر متصل می‌کند. اما متد post به صورت مخفی (و در واقع از طریق Http Request ها) برای این کار استفاده می‌کند.

از آن‌جا که متد get از آدرس برای ارسال داده‌ها استفاده می‌کند امکان نمایش محتویات فیلد‌های رمزی در نوار آدرس وجود دارد که این نشان دهنده عدم امنیت کافی در ارسال داده‌ها به طریق است. اما متد post به دلیل ارسال مخفی داده‌ها، امنیت کافی دارد.

تفاوت سوم در محدودیت حجم داده‌های ارسالی از طریق URL توسط مرورگرهاست. مرورگرها معمولاً در ارسال به این طریق محدودیت قائل می‌شوند که البته این مورد از مرورگر تا مرورگر متفاوت است. اما در روش ارسال داده‌ها از طریق متد post هیچ محدودیتی در حجم داده‌های ارسالی وجود ندارد.

یکی دیگر از تفاوت‌های این دو روش در سرعت ارسال داده‌هاست که معمولاً سرعت انجام این کار از طریق get کمی-بیشتر از post است.

¹ Server side

Query String رشته هایی هستند که به انتهای آدرس های URL افزوده می‌شوند و اغلب توسط فرم های HTML ایجاد می‌شوند. این رشته شامل تعدادی عبارت به صورت name=value (نام=رشته) بوده و برای ارسال داده های فرم به اسکریپتی بر روی سرور می‌دهند به کار می‌روند. در مثال زیر یک Query String را مشاهده می‌کنید:

```
?name=value&name2=value2&name3=value3...
```

کد ۹-۲

همانگونه که مشاهده می‌کنید در ابتدای Query String ها یک علامت ؟ قرار می‌گیرد و سپس تعدادی عبارت به صورت جفت های name=value نوشته می‌شود که هر کدام از آن ها با کاراکتر & از یکدیگر جدا می‌شوند.

در مثال زیر سه جفت name=value به کار رفته است:

```
http://www.mysite.com/processor.php?name=ali&family=akbari&age=20
```

کد ۹-۳

Enctype: این صفت نوع رمزگذاری^۱ اطلاعات فرم را مشخص می‌کند. که سه مقدار زیر را می‌تواند اختیار کند:



application/x-www-form-urlencoded v

هست مشخص می‌کند که داده های ارسالی به روش زیر می‌بایست رمزگذاری شوند:

تمامی کاراکتر های space به کاراکتر + و کاراکتر های خاص (مانند: &, ?, /,) و غیر الفبایی (در سیستم کدگذاری ascii) به کاراکتر های هگزادسیمال معادلشان تبدیل شوند.

multipart/form-data v

از این گزینه هنگامی استفاده می‌شود که داخل فرم یک عنصر file Uploading (<input type='file' />) وجود دارد.

text/plain v

از این گزینه می‌بایست هنگامی استفاده کرد که فرم به جای یک صفحه پردازشگر به یک ایمیل خاص ارسال می‌شود. در این گزینه فقط کاراکتر های space به + تبدیل شده و بقیه کاراکتر ها به همان شکلی که هستند ارسال می‌شوند.

برای ارسال یک فرم به یک آدرس ایمیل می‌بایست از پروتکل mailto: به شکل زیر استفاده نمود:

```
<form action='mailto:ahmadbadpey@gmail.com?subject=hello...'>
```

کد ۹-۴

¹ encoding

مجموعه ای از کاراکترها وجود دارند که در آدرس های اینترنتی وظیفه مشخصی بر عهده دارند و معنی خاصی می‌دهند. به همین دلیل نمی‌توان مستقیماً آن‌ها را به همان شکل در URL ها و Query String ها به کار برد (مانند کاراکتر ؟ که برای الحاق Query String ها به انتهای URL ها استفاده می‌شود یا کاراکتر & که برای جدا سازی جفت های name=value به کار می‌رود). به این کاراکترها، کاراکتر های خاص می‌گویند. بنابراین باید به جای آن از کد های خاصی استفاده کنیم. در زیر بعضی از کاراکتر های خاص به همراه کد های معادلشان را مشاهده می‌کنید:

Space()	%20
#	%23
&	%26
+	%2B
.	%2E
/	%2F
:	%3A
=	%3D
?	%3F
\	%5C
TAB	%90
!	%21

در ارسال یک فرم یک سری نکات وجود دارد که باید حتماً از آن‌ها پیروی کرد:

در صورتی که عنصری از نوع فایل در فرم وجود دارد می‌بایست صفت enctype را برابر multipart/form-data و صفت method را برابر post قرار دهیم.

در صورتی که فرم به یک ایمیل ارسال می‌شود می‌بایست enctype را برابر text/plain و صفت method را هم برابر post قرار دهیم.

تگ <label>



از این تگ برای ایجاد برچسب‌ها در کنار عناصر دیگر فرم استفاده می‌شود. در واقع label ها، همان متن های کوتاهی هستند که کاری را که یک عنصر می‌کند به کاربر نشان می‌دهند.

for: از این صفت برای نسبت دادن یک برچسب به یک عنصر خاص استفاده می‌شود. با این صفت می‌توان تمرکز بیشتری بر روی عناصر یک form داشت. این صفت ID یک عنصر از form را به عنوان مقدار می‌پذیرد.

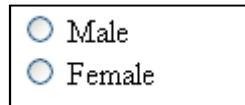


مثال زیر نحوه استفاده از این تگ را همراه با دکمه های رادیویی نشان می‌دهد:

```
<form>
  <input type="radio" name="type" id="male" />
  <label for="male">Male</label>
  <br />
  <input type="radio" name="type" id="female" />
  <label for="female">Female</label>
</form>
```

کد ۹-۵

حاصل اجرای کد فوق به صورت زیر خواهد بود:



شکل ۹-۲

تگ <input>



از این تگ برای ایجاد انواع و اقسام عناصر فرم همچون دکمه ها و فیلد های متنی و رمزی و... استفاده می شود. این تگ صفت های زیر را پشتیبانی می کند:

Name: برای نام گذاری عناصر استفاده می شود و می بایست در یک صفحه یکتا باشد.



Type: این صفت چندین مقدار می گیرد که هر کدام وظیفه ی خاصی دارند که در ادامه بررسی می کنیم:

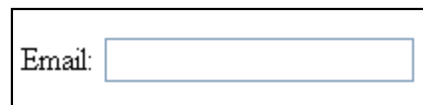


Text: برای ایجاد یک textfield ساده متنی به کار می رود.



```
<input type="text" name="email" />
```

کد ۹-۶



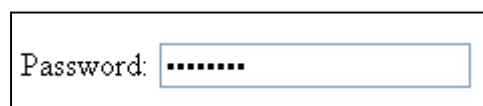
شکل ۹-۳

Password: برای ایجاد یک textfield رمزی به کار می رود.




```
<input type="password" name="pwd" />
```

کد ۹-۷

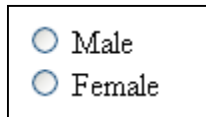


شکل ۹-۴


Radio : برای ایجاد دکمه‌های رادیویی^۱ به کار می‌رود. توجه داشته باشید که radio ها معمولا به صورت گروهی به کار می‌روند و می‌بایست نام آن‌ها در هر گروه یکسان بوده اما value های متفاوتی داشته باشند. کاربر فقط می‌تواند یک radio (گزینه) را از هر گروه انتخاب کند.

```
<input type="radio" name="sex" value="male" /> Male<br />
<input type="radio" name="sex" value="female" /> Female
```

کد ۹-۸

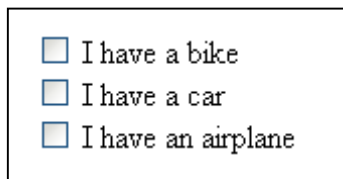


شکل ۹-۵


Checkbox : از این مقدار برای ایجاد checkbox ها استفاده می‌شود.

```
<input type="checkbox" name="vehicle" value="Bike" /> I have a bike
<br />
<input type="checkbox" name="vehicle" value="Car" /> I have a car
<br />
<input type="checkbox" name="vehicle" value="Airplane" /> I have an
airplane
```

کد ۹-۹



شکل ۹-۶


Submit : برای ایجاد دکمه‌های از نوع submit (ارسال) استفاده می‌شود. در واقع این نوع دکمه‌ها وظیفه ارسال فرم به صفحه پردازشگر را دارند.


```
<input type="submit" value="Submit" />
```

کد ۹-۱۰



شکل ۹-۷

Reset : برای ایجاد دکمه‌های پاک‌کننده^۲ ی form به کار می‌روند. این نوع دکمه‌ها محتویات فرم را به مقدار اولیه (پیشفرض) بر می‌گرداند.

Button : از این مقدار برای ایجاد دکمه‌هایی که مثل دو نوع دکمه ی قبلی رفتاری پیش فرض از خود ندارند استفاده می‌شود.

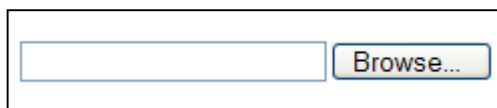
^۱ Radio button

^۲ reset

File: از این مقدار برای ایجاد امکان آپلود فایل برای کاربران استفاده می‌شود. در صورت استفاده از این مقدار یک کادر متنی به همراه دکمه ای با عنوان **Browse** ایجاد خواهد شد که کاربر خواهد توانست با استفاده از آن فایلی را از کامپیوتر خود انتخاب کند.

```
<input type="file" />
```

کد ۹-۱۱



شکل ۹-۸

hidden: گاهی اوقات طراح فرم می‌خواهد مقداری را به دور از چشم کاربر و بدون دخالت آن برای صفحه پردازشگر ارسال کند. در این صورت می‌توان از مقدار **hidden** استفاده کرد.

image: از این مقدار برای ایجاد دکمه های تصویری از نوع **submit** استفاده می‌شود. در واقع با این مقدار می‌توان تصویری ایجاد کرد که کاربر با کلیک بر روی آن می‌تواند **form** را ارسال کند. باید به این نکته توجه داشت که زمانی که از مقدار **image** برای صفت **type** استفاده می‌شود می‌توانیم از تمامی صفت های **img** برای کنترل خصوصیات تصویر استفاده کنیم.

```
<input type="image" src="submit.gif" alt="Submit" />
```

کد ۹-۱۲



شکل ۹-۹


نکته جالب اینکه با استفاده از این مقدار می‌توان مختصات نقطه ای از عکس را که کاربر بر روی آن کلیک کرده است را به **server** ارسال کرد. در این صورت دو مقدار **X** (فاصله نقطه از سمت چپ تصویر) و **Y** (فاصله نقطه از بالای تصویر) به صورت زیر برای صفحه پردازشگر ارسال می‌شوند: **label.x** و **label.y**. در این حالت **label** نام دکمه خواهد بود.

email: از این مقدار برای آن دسته از فیلد هایی که می‌بایست حاوی یک آدرس ایمیل معتبر باشند استفاده می‌شود. مقدار این فیلد قبل از ارسال فرم خطایابی خواهد شد و چنانچه آدرس ایمیل معتبر وارد نشده باشد یک پیام خطا نمایش داده می‌شود:

```
E-mail: <input type="email" name="user_email" />
```

کد ۹-۱۳

تصویر زیر نحوه نمایش پیام خطا در مرورگر Chrome را نشان می‌دهد:



شکل ۹-۱۰

URL: از این مقدار برای آن دسته از فیلد هایی که می‌بایست حاوی یک آدرس اینترنتی معتبر باشند استفاده می‌شود. مقدار این فیلد قبل از ارسال فرم خطایابی خواهد شد و چنانچه آدرس معتبر وارد نشده باشد یک پیغام خطا نمایش داده می‌شود:

```
Homepage: <input type="url" name="user_url" />
```

کد ۹-۱۴

تصویر زیر نحوه نمایش پیغام خطا در مرورگر Chrome را نشان می‌دهد:



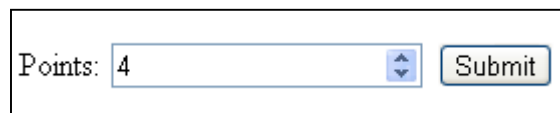
شکل ۹-۱۱

number: از این مقدار برای آن دسته از فیلد هایی که می‌بایست حاوی یک مقدار عددی باشند استفاده می‌شود. ما همچنین می‌توانیم محدودیت هایی روی اعداد قابل قبول نیز اعمال کنیم:

```
Points: <input type="number" name="points" min="1" max="10" step="3" value="4" />
```

کد ۹-۱۵

حاصل اجرای کد فوق در مرورگر chrome به صورت زیر است:



شکل ۹-۱۲

range: از این مقدار برای آن دسته از فیلد هایی که می‌توانند حاوی مقدار مشخصی از بازه خاصی از اعداد باشند استفاده می‌شود. ما همچنین می‌توانیم محدودیت هایی روی اعداد قابل قبول نیز اعمال کنیم:

```
Points: <input type="range" name="points" min="1" max="10" />
<input type="submit" />
```

کد ۹-۱۶

حاصل اجرای کد فوق در مرورگر chrome به صورت زیر است:

شکل ۹-۱۵

سایر مقادیری که می‌توان برای انتخاب زمان و تاریخ استفاده کرد شامل مقادیری همچون `month`، `week`، `time`، `datetime` و `datetime-local` می‌باشد.

Search V: از این مقدار همانطور که از نام آن نیز مشخص است برای ایجاد فیلدهای جستجو، همانند جستجو در سایت و جستجوی گوگل استفاده می‌شود. این نوع فیلدها همانند کادرهای متنی یک خطی نمایش داده می‌شوند:

```
Search In Site: <input type="search" name="search_box" />
```

کد ۹-۱۹

size: از این صفت هنگامی که صفت `type` برابر یکی از مقادیر `text`، `password` یا `file` باشد استفاده می‌شود و به وسیله آن عرض این نوع عناصر را می‌توان کنترل کرد. این صفت مقداری عددی را می‌پذیرد و به اندازه همان تعداد کاراکتر به این نوع عناصر عرض می‌دهد. به عنوان مثال اگر برای این صفت ۵۰ را در نظر بگیریم عنصر مورد نظر به اندازه ۵۰ کاراکتر عرض خواهد داشت.

maxlength: این مقدار هم عددی صحیح را می‌پذیرد و حداکثر تعداد کاراکتری را که کاربر می‌تواند در عناصری از نوع `text`، `password` یا `file` وارد کند را مشخص می‌کند.

checked: هنگامی استفاده می‌شود که صفت `type` برابر `checkbox` یا `radio` باشد. این صفت بولین موجب می‌شود که عناصر از این نوع به صورت پیش فرض به حالت انتخاب شده در آیند.

```
<input type="checkbox" name="vehicle" value="Bike" /> I have a bike
<br />
<input type="checkbox" name="vehicle" value="Car" checked="checked" />
I have a car
```

کد ۹-۲۰

I have a bike
 I have a car

شکل ۹-۱۶

src: این صفت آدرس عکسی که قرار است به عنوان دکمه submit ایجاد شود را مشخص می‌کند. از این صفت هنگامی که مقدار صفت type برابر image است استفاده می‌شود.

accept: از این صفت هنگامی که صفت type برابر file باشد استفاده می‌شود و می‌توان نوع فایل‌هایی که کاربر از کامپیوتر خود برای آپلود انتخاب کند را مشخص می‌کند. این صفت لیستی از MIMEType فایل‌های مورد نظر را به عنوان مقدار می‌پذیرد و به عنوان مثال در صورتی که بخواهیم کاربر فقط بتواند فایل‌های از نوع gif یا jpg را انتخاب کند به صورت زیر استفاده می‌کنیم:

```
<input type='file' accept='image/gif,image/jpeg' />
```

کد ۹-۲۱

accesskey: توسط این صفت می‌توان shortcut (میانبری) برای عنصر فرم تعریف کرد. برای اطلاعات بیشتر به توضیحات همین صفت در تگ a مراجعه نمایید.

tabindex: برای اطلاعات بیشتر به توضیحات همین صفت در تگ a مراجعه نمایید.

readonly: از این صفت برای فقط خواندنی کردن عناصر فرم استفاده می‌شود. این صفت که جزء صفت‌های boolean است در صورتی که برای عناصر فرم استفاده شود کاربر امکان تغییر محتویات آن عنصر را نخواهد داشت.

disabled: این صفت boolean عناصر را به حالت غیر فعال در می‌آورد.

صفات readonly و disabled تفاوت‌ها و شباهت‌هایی دارند که در ادامه به مهمترین آن‌ها اشاره می‌کنیم.

readonly	disabled	
امکان focus بر روی عنصر	عدم امکان focus بر روی عنصر	تفاوت‌ها
امکان tabbing	عدم امکان tabbing	
محتویاتش به سرور ارسال می‌شود	محتویاتش به سرور ارسال نمی‌شود	
امکان تغییر محتویاتشان وجود ندارد		شباهت‌ها

صفت‌های جدید برای تگ‌های input

همانند مقادیر جدیدی که برای صفت type در تگ input در HTML 5 ارائه شده است در این نسخه صفت‌های جدیدی نیز برای این تگ در نظر گرفته شده است که در ادامه به مهمترین آن‌ها خواهیم پرداخت:



autofocus: اعمال این صفت بولین بر روی یک تگ `input` موجب می‌شود تا در هنگام بارگزاری صفحه `input` به صورت پیش فرض بر روی آن قرار گیرد. به این نکته نیز دقت کنید که در هر صفحه فقط یک تگ `input` می‌تواند این صفت را اختیار کند:

```
First name:<input type="text" name="fname" autofocus="autofocus" /><br />
Last name: <input type="text" name="lname" /><br />
<input type="submit" />
```

کد ۹-۲۲

شکل ۹-۱۷



placeholder: یکی از متداولترین کارهایی که در مورد کادرهای متنی انجام می‌شود قرار دادن متنی به صورت پیش فرض در داخل آن هاست که در هنگام اعمال `focus` بر روی آن‌ها، پاک می‌شوند. این کار تا قبل از این به دشواری و با استفاده از جاوااسکریپت انجام می‌شد. اما اکنون با ارائه صفت `placeholder` در `HTML5` این کار به آسانی انجام می‌شود. این صفت مقداری که قرار است به صورت پیش فرض در داخل کادر متنی نمایش داده شود را می‌پذیرد. از این صفت معمولاً در کادرهای جستجو استفاده می‌شود:

```
<input type="search" name="user_search" placeholder="Search in Google" />
<input type="submit" />
```

کد ۹-۲۳

شکل ۹-۱۸

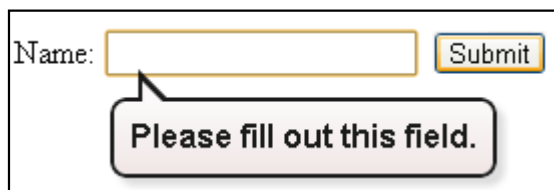


required: از این صفت بولین برای آن دسته از عناصری که نمی‌توان آن‌ها را خالی رها کرد و در واقع وارد کردن آن الزامی است استفاده می‌شود. از این صفت می‌توان در تگ‌های `textarea` و اغلب تگ‌های `input` (به جز آنهایی که صفت `type` شان برابر `hidden`، `image` و بعضی از دکمه‌ها همچون دکمه‌های `submit`) قابل استفاده است. مرورگرهایی که از این صفت پشتیبانی می‌کنند در صورت خالی رها کردن فیلدهایی که این صفت را اختیار کرده اند پیغام خطا نمایش می‌دهند:

```
Name: <input type="text" name="usr_name" required="required" />
<input type="submit" />
```

کد ۹-۲۴

تصویر زیر نمایش پیغام خطا برای فیلد فوق در مرورگر Chrome را نمایش می‌دهد:



شکل ۹-۱۹

ایجاد combobox و listbox ها

از جمله مهمترین و البته پر کاربردترین عناصر فرم، combobox ها و listbox ها هستند. این عناصر با استفاده از ترکیب تگ های select و option ایجاد می‌شوند. همچنین از تگ optgroup در داخل تگ select برای گروه بندی آیتم ها عناصر فوق الذکر استفاده می‌شود.

تگ <select>

برای ایجاد عناصر combobox یا listbox استفاده می‌شود. این تگ از صفت های زیر پشتیبانی می‌کند.

name: نامی برای عنصر انتخاب می‌کند

size: برای ایجاد listbox ها استفاده می‌شود و عددی صحیح را می‌پذیرد و تعداد آیتم های نمایشی در listbox را مشخص می‌کند.

multiple: این صفت boolean امکان انتخاب چندین گزینه در یک listbox را برای کاربر فراهم می‌کند. برای این کار کاربر باید دکمه Ctrl صفحه کلید را نگه دارد.

تگ <option>

همیشه داخل تگ <select> استفاده می‌شود و آیتم های داخل آن را مشخص می‌کند. مقداری که بین این تگ قرار می‌گیرد عنوان هر آیتم خواهد بود. صفت های این تگ به شرح زیر می‌باشد:

value: مقداری را برای هر آیتم مشخص می‌کند. value مقداری خواهد بود که در صورت انتخاب آیتم توسط کاربر برای صفحه پردازشگر ارسال می‌شود. زمانی که این صفت تعیین نشود محتوای عنصر option به عنوان value در نظر گرفته می‌شود.

selected: در صورت استفاده از این صفت boolean برای یک option آن آیتم به صورت پیش فرض به حالت انتخاب شده در خواهد آمد.

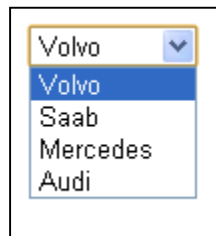
در یک combobox فقط یک آیتم می‌تواند این صفت را اختیار کند ولی در یک listbox چندین آیتم می‌تواند این صفت را داشته باشند.

مثال زیر نحوه ایجاد یک combobox را نشان می‌دهد:

```
<select name='cars' id="cars">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="mercedes">Mercedes</option>
  <option value="audi">Audi</option>
</select>
```

کد ۹-۲۵

حاصل اجرای کد فوق به صورت زیر خواهد بود:



شکل ۹-۲۰

حال اگر به دستورات فوق صفت های multiple و size را نیز اضافه کنیم یک listbox ایجاد خواهد شد:

```
<select multiple="multiple" size="2">
  <option value="volvo" selected="selected">Volvo</option>
  <option value="saab">Saab</option>
  <option value="mercedes" selected="selected">Mercedes</option>
  <option value="audi">Audi</option>
</select>
```

کد ۹-۲۶

حاصل اجرای کد فوق به صورت زیر خواهد بود:



شکل ۹-۲۱

تگ <optgroup>



از این تگ برای گروه بندی option های listbox یا combobox استفاده می‌شود. برای این کار می‌بایست option های مورد نظر را داخل تگ باز و بسته optgroup قرار دهیم. صفت های این تگ به شرح زیر است:

label: عنوان هر گروه را مشخص می‌کند.



کد زیر نحوه گروه بندی عناصر یک combobox را با استفاده از این تگ نمایش می دهد:

```
<select>
  <optgroup label="Swedish Cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
  </optgroup>
  <optgroup label="German Cars">
    <option value="mercedes">Mercedes</option>
    <option value="audi">Audi</option>
  </optgroup>
</select>
```

کد ۹-۲۷

حاصل اجرای کد فوق به صورت زیر خواهد بود:



شکل ۹-۲۲

ایجاد لیست های داده^۱

یکی از امکانات جدید فراهم شده در HTML5 در رابطه با عناصر فرم امکان ایجاد لیست های داده است. لیست های داده بسیار شبیه combobox ها هستند اما تفاوت آن ها در این است که بر خلاف combobox که امکان انتخاب یک گزینه از میان گزینه های از پیش تعریف شده را میدهد در لیست های داده امکان وارد کردن گزینه مورد نظر کاربر که در میان گزینه های از پیش تعریف شده نیست وجود دارد. برای ایجاد یک datalist نیاز به یک کادر متنی و یک تگ <datalist> خواهیم داشت. سپس کافی است نامی برای تگ datalist از طریق صفت id تعیین کرده و صفت جدید list را در تگ input مربوطه برابر آن قرار دهید. به کد زیر دقت کنید:

```
<input list="cars" />
<datalist id="cars">
  <option value="BMW">
  <option value="Ford">
  <option value="Volvo">
</datalist>
```

کد ۹-۲۸

حاصل اجرای کد فوق در مرورگر Opera به صورت زیر خواهد بود:

^۱ datalist


samand
BMW
Ford
Volvo


شکل ۹-۲۳


ایجاد کادر های متنی چند خطی


تگ <textarea>

از این تگ برای ایجاد نواحی متنی چند خطی استفاده می‌شود و صفت های زیر را می‌پذیرد:

Rows: عددی صحیح را به عنوان مقدار می‌پذیرد و به همان تعداد **سطر** به ناحیه متنی ارتفاع می‌دهد. 

Cols: عددی صحیح را به عنوان مقدار می‌پذیرد و به همان تعداد **کاراکتر** به ناحیه متنی عرض می‌دهد. 

Readonly: قبلا توضیح داده شده است 

Disabled: قبلا توضیح داده شده است 

دستورات زیر را در نظر بگیرید:

```
<textarea rows="2" cols="20">
  At W3Schools you will find all the Web-building tutorials you need, from
  basic HTML to advanced XML, SQL, ASP, and PHP.
</textarea>
```

کد ۹-۲۹

حاصل به صورت زیر خواهد بود:


At W3Schools you will find all the Web-building tutorials you need, from basic HTML to advanced XML, SQL, ASP, and PHP.


شکل ۹-۲۴

ایجاد انواع دکمه ها

تگ <button>

از این تگ برای ایجاد هر سه نوع دکمه ای که قبلا به آن ها اشاره شد می‌توان استفاده نمود. این تگ از صفت های زیر پشتیبانی می‌کند:

name: نامی برای دکمه تعیین می‌کند 

value: مقداری که برای صفحه پردازشگر ارسال می‌شود را مشخص می‌کند 

type: نوع دکمه که ایجاد می‌شود مشخص می‌کند و می‌تواند سه مقدار زیر را بگیرد:



submit: یک دکمه از نوع submit برای ارسال فرم ایجاد می‌کند



reset: یک دکمه از نوع reset به منظور پاک کردن عناصر فرم ایجاد می‌کند.



button: یک دکمه از نوع ساده که رفتار پیش فرضی از خود ندارد ایجاد می‌کند



در این نوع دکمه متنی که داخل تگ باز و بسته button قرار می‌گیرد عنوان دکمه خواهد بود:

```
<button type="button">Click Me!</button>
```

کد ۹-۳۰



شکل ۹-۲۵

سازمان دهی عناصر فرم

تگ **<fieldset>**



از این تگ برای نظم دهی و گروه بندی دیگر عناصر فرم استفاده می‌شود. این تگ فقط صفت های عمومی را دارد.

تگ **<legend>**



این تگ همیشه داخل تگ fieldset استفاده می‌شود و عنوان آن را مشخص می‌کند و از صفت زیر پشتیبانی می‌کند:

align: موقعیت عنوان را مشخص می‌کند و یکی از مقادیر زیر را می‌پذیرد:



left: عنوان را در بالا سمت چپ قرار می‌دهد



center: عنوان را در بالا وسط قرار می‌دهد



right: عنوان را در بالا سمت راست قرار می‌دهد.



دستور زیر نحوه استفاده از تگ های fieldset و legend را نشان می‌دهد:

```
<fieldset>
  <legend>Personalia:</legend>
  Name: <input type="text" size="30" /><br />
  Email: <input type="text" size="30" /><br />
  Date of birth: <input type="text" size="10" />
</fieldset>
```

کد ۹-۳۱

حاصل به صورت زیر خواهد بود:

Personalia:

Name:

Email:

Date of birth:

شکل ۹-۲۶

ایجاد نوار های پیشرفت^۱

یکی از عناصر ارائه شده در HTML5 نوار پردازش است. از این عنصر معمولا برای نمایش روند پردازش عملیات (مخصوصا عملیات دانلود و آپلود) استفاده می‌شود. برای ایجاد این عنصر از تگی به نام `<progress>` استفاده می‌شود. این تگ صفت های `max` برای مشخص کردن مقدار نهایی و صفت `value` برای مشخص کردن مقدار کنونی نوار را مشخص می‌کند. به مثال زیر دقت کنید:

The object's downloading progress:

```
<progress value="22" max="100">
```

```
</progress>
```

کد ۹-۳۲

دستور فوق در مرورگر chrome به صورت زیر نمایش داده می‌شود:

The object's downloading progress:

شکل ۹-۲۷

^۱ progress bar

فصل دهم

عناصر چند رسانه ای در صفحات وب

یکی از نکاتی که وب را تا این حد پرترفدار کرده است این حقیقت است که شما می توانید تصاویر گرافیکی ، فایل های صوتی ، تصویر متحرک و فیلم ها را به صفحات وبتان اضافه کنید. اگر چه در گذشته اندازه این فایل ها سبب محدود شدن استفاده شان می شد، تکنولوژی های جدیدتر مانند جریان های صوتی و ویدیویی در کنار اتصالات با پهنای باند زیاد در ها را به روی صفحات وب چند رسانه ای باز کرده است.

ما در این فصل ابتدا با انواع عناصر چند رسانه ای و فرمت های قابل استفاده آن ها در صفحات وب آشنا شده و سپس به نحوه گنجاندن آن ها در صفحات اینترنتی خواهیم پرداخت.

تعریف عناصر چندرسانه ای

اگر بخواهیم عناصر چندرسانه ای را تعریف کنیم باید بگوییم که کلا هر آنچه که در یک صفحه وب قابل دیدن یا شنیدن باشد یک عنصر چندرسانه ای خواهد بود. مانند متن ها، کتاب های الکترونیکی^۱، تصاویر، موزیک ها، صدا ها فیلم ها و... عناصر چند رسانه ای در فرمت های مختلفی وجود دارند که مرور گر های امروزی از تعدادی از آنها پشتیبانی می کنند. یک مرورگر فقط توانایی نشان دادن متن و تعداد معدودی از تصاویر را دارد اما انواع فراوان و مختلفی از فایلها روی وب وجود دارد از جمله فایلهای ویدیویی، فایل های PDF، انیمیشن FLASH و... برای اینکه یک مرورگر بتواند سایر انواع فایل ها را نمایش دهد به برنامه های کمکی، به نام **برنامه های پخش کننده** نیاز دارند. برنامه های پخش کننده به دو شکل خارجی و مجزا که به صورت برنامه های جداگانه روی کامپیوتر مراجعه کنندگان اجرا می شوند و همچنین برنامه های پخش کننده plugin که درون صفحه وب در پنجره مرورگر کار می کنند وجود دارند. متداول ترین plugin ها پخش کننده های flash و shockwave از شرکت ماکرو مدیا، پخش کننده Quicktime از شرکت Apple، Windows Media Player از شرکت مایکروسافت و Acrobat reader از شرکت Adobe می باشند.

بررسی انواع عناصر چندرسانه ای

انواع عناصر چند رسانه ای که می توان از طریق HTML آن ها را در صفحات وب قرار داد به دسته های زیر تقسیم بندی می شوند:

فایل های تصویری

یکی از رایج ترین عناصر چند رسانه ای که در بسیاری از صفحات اجرا می شوند فایل های تصویری یا همان عکس ها هستند. معمولا این عناصر به صورت خودکار توسط انواع مرورگر ها شناخته می شوند. از رایج ترین فرمت های تصاویر که در صفحات قرار می گیرند می توان به .jpg، .gif، و .png اشاره نمود.

فایل های متنی

این نوع فایل ها جزء اولین انواع بودند که توسط مرورگر ها پشتیبانی شدند. از این دسته از فایل ها می توان به فایل های html و PDF ها اشاره نمود. مرورگر ها برای نمایش فایل های PDF به نرم افزار کمکی Acrobat Reader نیاز دارند.

فایل های صوتی

فایل های صوتی کاربردی در صفحات وب در فرمت های مختلفی ذخیره می شوند که به بعضی از آن ها اشاره می شود:

- **فرمت MIDI**: این فرمت برای اولین بار در سال ۱۹۸۲ ابداع شد و به عنوان یکی از ابتدایی ترین فرمت های قابل استفاده در صفحات وب نیز شناخته می شود. یکی از ویژگی های منحصر به فرد این فرمت این است که این نوع فقط قادر به ضبط و نگهداری نت های موسیقی است و امکان ضبط صدا ها بر روی این فرمت وجود ندارد. یکی از مزیت های این فرمت حجم بسیار پایین آن است. این فرمت در بسیاری از سیستم عامل ها و مرورگر ها به خوبی پشتیبانی می شود. فایل های از این نوع معمولا با یکی از پسوند های .mid یا .midi. ذخیره می شوند.

- **فرمت Wave:** این فرمت توسط شرکت مایکروسافت ارائه شده است. از مزایای این نوع می‌توان به کیفیت بسیار بالای آن اشاره نمود اما در مقابل حجم زیادی هم دارند. این فرمت فقط بر روی سیستم عامل Windows و بر روی تمامی مرورگرها قابل اجراست. فایل‌های این نوع در پسوند‌های wav. ذخیره می‌شوند.
 - **MP3(MPEG):** فایل‌های MP3 در حقیقت همان فایل‌های MPEG هستند. اما فایل‌های MPEG به صورت اختصاصی برای فایل‌های ویدیویی معرفی و توسعه پیدا کرده‌اند. در واقع ما می‌توانیم بگوییم که فایل‌های MP3 صدای جدا شده از فایل‌های ویدیویی MPEG هستند!
 - **MP3** یکی از محبوبترین و پر کاربردترین نوع فایل‌های صوتی است که توسط همه مرورگرها و سیستم‌عامل‌ها به خوبی پشتیبانی می‌شود. از مزیت‌های این نوع می‌توان به کیفیت برابر با wave اما حجم کمتر اشاره نمود. فایل‌های این نوع در پسوند‌های MP3 یا MPGA. ذخیره می‌شوند.
- از بین فرمت‌های ذکر شده می‌توان از wave یا MP3 به عنوان بهترین گزینه برای فایل‌های صوتی نام برد. از دیگر فرمت‌های صوتی که ممکن است در صفحات استفاده شوند اما کاربرد کمتری دارند و پشتیبانی خوبی از آن‌ها در همه سیستم‌عامل‌ها یا مرورگرها نشود می‌توان به موارد زیر اشاره نمود:

.rm ,.ram ,.aif ,.aiff ,.snd

فایل‌های ویدیویی

این دسته از فایل‌ها در فرمت‌های گوناگونی عرضه می‌شوند که در زیر به برخی از آن‌ها اشاره می‌شود:

- **AVI:** اولین بار توسط مایکروسافت ارائه شد. این فرمت یکی از رایج‌ترین فرمت‌های ویدیویی مورد استفاده در صفحات وب هستند که توسط بسیاری از سیستم‌عامل‌ها و مرورگرها پشتیبانی می‌شوند. فایل‌های این نوع در پسوند avi ذخیره می‌شوند.
- **WMV:** یکی از رایج‌ترین فرمت‌های ویدیویی مورد استفاده در صفحات وب است که البته فقط بر روی سیستم عامل ویندوز قابل اجراست و برای اجرا بر روی سایر سیستم‌عامل‌ها نیاز به نصب برنامه‌های کمکی ویژه دارد. فایل‌های این دسته با پسوند wmv. ذخیره می‌شوند.
- **MPEG:** یکی از محبوبترین فرمت‌های ویدیویی برای صفحات وب است. این فرمت در تمامی سیستم‌عامل‌ها و مرورگرها به خوبی پشتیبانی می‌شود. فایل‌های این نوع در پسوند‌های mpg. یا mpeg. ذخیره می‌شوند.
- **QuickTime:** اولین بار توسط شرکت Apple ابداع و عرضه شد. فایل‌های این فرمت برای اجرا نیاز به برنامه کمکی quicktime دارند. فایل‌های این دسته در پسوند‌های mov. یا 3gp. ذخیره می‌شوند.
- **Shockwave(Flash):** شرکت Macromedia این نوع فرمت را برای اولین بار ابداع کرد. فایل‌های این نوع برای اجرا نیاز به برنامه کمکی Adobe Flash Player دارند. این برنامه معمولاً بر روی آخرین نسخه مرورگرها به صورت پیش فرض نصب می‌باشد. فایل‌های این نوع فقط با پسوند swf. ذخیره می‌شوند.
- **اپلت‌های جاوا^۱:** یکی دیگر از عناصری که امکان گنجاندن آن‌ها در صفحات وب وجود دارد اپلت‌های جاوا است که البته برای اجرا در مرورگر نیاز به نصب java از قبل داریم.

¹ Audio Video Interleave

تگ Object



حال که با بسیاری از فرمت‌ها ی قابل استفاده عناصر چند رسانه ای در صفحات وب آشنا شدیم در ادامه با طریقه گنجاندن آنها از طریق تگ های مشخص می‌پردازیم.

استفاده از لینک ساده به فایل

ساده ترین راه برای گنجاندن انواع فایل های چند رسانه ای استفاده از لینکی است که به یکی از انواع فوق اشاره دارد. در این صورت مرورگر ها با استفاده از یک برنامه ی کمکی^۲ و دریک صفحه جدید فایل را باز خواهند کرد.

استفاده از تگ Object

از این تگ برای گنجاندن انواع فایل های چند رسانه ای استفاده می‌شود که در ادامه صفت های آن را بررسی می‌کنیم:

width



height



border



hspace



vspace



align



name



data: مسیر کامل نگه داری فایل مورد نظر برای نمایش را مشخص کند.



archive: لیستی از آدرس هایی که نگه دارنده منابع و فایل های مربوط به تگ object است را به عنوان مقدار



می‌پذیرد

classid: مشخص کننده classid برای برنامه اجرا کننده فایل چند رسانه ای است که در registry ثبت می‌شود.



type: نوع MIMEType فایل را مشخص می‌کند.



codebase: موقعیت یا آدرس جایی که کاربر می‌تواند برنامه plugin مورد نیاز برای پخش عنصر چندرسانه ای را



دریافت نماید را مشخص می‌کند.

standby: متنی را به عنوان مقدار می‌پذیرد و آن را در هنگام بارگذاری فایل نمایش می‌دهد.



declare: در صورتی که این صفت boolean تنظیم شود، مشخص می‌کند که فایل مورد نظر فقط باید معرفی



شود و تا زمانی که نیاز نیست اجرا یا نصب نشود.

¹ Java applet

² Helper Application

تگ param

معمولا تگ object همراه با تگی به نام param که تنظیمات زمان اجرای برنامه را کنترل می‌کند به کار می‌رود. صفت های تگ param به شرح زیر است:

name: یک نام مشخص برای پارامتر تعیین می‌کند.

value: مقداری را برای پارامتری که در صفت name آمده است را مشخص می‌کند.

مقادیری که می‌توان برای صفت name تعیین نمود بسته به اینکه تگ object چه نوع فایلی را نمایش می‌دهد متفاوت است مثلا اگر فایل ما یک فایل swf باشد می‌توانیم پارامترها و مقادیر زیر را برای آن تعیین نماییم:

src: آدرس فایل مورد نظر برای نمایش را مشخص می‌کند. Internet Explorer از این گزینه برای پیدا کردن فایل مورد نظر برای نمایش استفاده می‌کند.

quality: این پارامتر کیفیت نمایش فایل فلش را کنترل می‌کند و می‌تواند یکی از مقادیر high , medium , low را بپذیرد.

loop: تکرار نمایش فایل فلش را کنترل می‌کند که می‌تواند یکی از مقادیر true برای ادامه پیدا کردن نمایش و false برای عدم تکرار نمایش بعد از پایان اجرا را بپذیرد.

اگر فایل نمایشی ما یک فایل صوتی، تصویری یا ویدیویی باشد می‌توان پارامترهای زیر را برای آن ها تعیین نمود:

autoplay: اجرای خودکار فایل را مشخص می‌کند و می‌تواند یکی از مقادیر true (اجرای خودکار) یا false (عدم اجرای خودکار) را بپذیرد.

controller: نمایش یا عدم نمایش کنترلگرها را مشخص می‌کند و می‌تواند یکی از مقادیر true یا false را بپذیرد.

scale: میزان بزرگنمایی فیلم را کنترل می‌کند و می‌تواند یک عدد صحیح مثلا ۵ باشد.

loop: تکرار نمایش فایل را کنترل می‌کند که می‌تواند یکی از مقادیر true برای ادامه پیدا کردن نمایش و false برای عدم تکرار نمایش بعد از پایان اجرا را بپذیرد.

یک مثال از کاربرد تگ های object و param را که یک فایل فلش را در صفحه نمایش خواهد داد در زیر مشاهده کنید:

```
<object data='objects/fc.swf' width='120' height='240'
type='application/x-shockwave-flash' border='5'
classid='clsid:D27CDB6E-AE6D-11cf-96B8-444553540000'>
  <param name='src' value='objects/fc.swf' />
  <param name='quality' value='low' />
  <param name='loop' value='false' />
</object>
```

راه های دیگر برای استفاده از عناصر چند رسانه ای

تگ <embed>



استفاده از این تگ بسیار آسان تر از تگ object است. از این تگ برای قرار دادن انواع فایل های چند رسانه ای می توان استفاده نمود. صفت های این تگ به شرح زیر است:

align 

alt 

border 


width 


height 


hspace 


vspace 


name 


src: آدرس منبع را مشخص می کند. 

autostart: یکی از مقادیر true یا false را به عنوان مقدار می پذیرد و مشخص می کند آیا عنصر چند رسانه ای می تواند به صورت خودکار اجرا شود یا خیر. true به معنی اجرای خودکار خواهد بود. 

hidden: یکی از مقادیر true یا false را به عنوان مقدار می پذیرد و مشخص می کند آیا عنصر چند رسانه ای نمایش داده شود یا خیر. false به معنی مخفی شدن عنصر است. 

loop: یکی از مقادیر true یا false را به عنوان مقدار می پذیرد و مشخص می کند آیا پخش عنصر چند رسانه ای پس از پایان ادامه پیدا کند یا خیر. مقدار true به معنی ادامه پخش عنصر خواهد بود. 

type: نوع MIMEType فایل مورد نظر را مشخص می کند. 

pluginspage: همانند صفت codebase در تگ object عمل می کند (فقط در مرورگر NetScape) 

برخی از مرورگر ها همچون FireFox و Opera از تگ Object برای قرار دادن فایل های ویدیویی و صوتی در صفحات وب پشتیبانی نمی کنند و فقط از تگ embed برای این کار استفاده می کنند.

اما یکی از مزیت های تگ object امکان قرار دادن سایر تگ ها از جمله تگ embed در داخل آن است. با این روش امکان قرار دادن این نوع فایل ها در تمامی مرورگر ها امکان پذیر می شود. مرورگر هایی که از object پشتیبانی می کنند آن را به همراه پارامتر های داخلش اجرا کرده و از تگ embed داخل آن چشم پوشی می کند. اما مرورگر های ناسازگار با تگ object از این تگ و تمامی پارامتر های داخلش صرف نظر کرده و در عوض تگ embed داخل آن را به طور کامل اجرا می کنند.

به عنوان مثال می توان کد ۱۰-۱ را از این روش به صورت زیر باز نویسی کنیم:

```
<object data='objects/fc.swf' width='120' height='240'
type='application/x-shockwave-flash' border='5'
classid='clsid:D27CDB6E-AE6D-11cf-96B8-444553540000'>
  <param name='src' value='objects/fc.swf' />
  <param name='quality' value='low' />
  <param name='loop' value='false' />
  <embed src='objects/fc.swf' width='120' height='240'></embed>
</object>
```

کد ۱۰-۲

تگ <bgsound>



از این تگ که فقط در مرورگر Internet Explorer پشتیبانی می‌شود می‌توان برای قرار دادن یک فایل صوتی در پس زمینه صفحه استفاده نمود تا این فایل به محض بارگذاری صفحه به صورت خودکار پخش شود. نکته اینک این عنصر جزء عناصر تهی بوده و حتما باید در داخل تگ <head> صفحه استفاده شود. این تگ از صفت های زیر پشتیبانی می‌کند:

src: آدرس فایل مورد نظر برای پخش را مشخص می‌کند.



loop: تعداد تکرار پخش صدا را کنترل می‌کند.



volume: برای کنترل میزان صدای فایل پخش شونده استفاده می‌شود و مقداری بین 0 تا 10000- را به عنوان مقدار می‌پذیرد. در واقع مقدار این صفت مقدار کاهش صدا را کنترل می‌کند. مثلا مقداری برابر 5000- به معنی نصف کردن صدا نبوده و به معنی کاهش 5000 واحد از کل صدای سیستم است.



balance: تراز پخش صدا بین بلندگوهای چپ و راست را کنترل می‌کند و مقداری بین 10000- تا 10000+ را به عنوان مقدار می‌پذیرد. هر چه این مقدار به 10000+ نزدیکتر باشد میزان صدای پخش شده از بلندگوی سمت راست و هر چه این مقدار به 10000- نزدیکتر باشد میزان صدای پخش شده از بلندگوی سمت چپ بیشتر خواهد بود.



یکی دیگر از راه های قرار دادن موزیک های پس زمینه استفاده از تگ embed است که در این حالت باید حتما صفت hidden را برابر true قرار دهیم. این ترفند به خوبی در تمامی مرورگر ها کار می‌کند.

استفاده از ویژگی های HTML 5 برای نمایش صدا و ویدئو

HTML5 روش ها و تگ های جدیدی برای قرار دادن عناصر چند رسانه ای در صفحات وب را فراهم کرده است. تگ video برای قرار دادن فایل های ویدیویی و تگ audio برای قرار دادن فایل های صوتی در این نسخه ارائه شده اند. مزیت های این دو تگ نسبت به تگ هایی همچون object و embed این است که مرورگر هایی که از این دو تگ جدید پشتیبانی می‌کنند قادرند اینگونه فایل های صوتی و ویدیویی را بدون نیاز به پلاگین ها اجرا کنند.

تگ <video>

ابتدا به بررسی تگ video می‌پردازیم. این تگ از صفت های زیر پشتیبانی می‌کند:

audio: وضعیت پیش فرض صدای فایل ویدیویی را مشخص می‌کند و در صورتی که آن را برابر مقدار muted قرار دهید صدای فایل از همان ابتدای اجرا قطع خواهد بود.

autoplay: این صفت بولین همانطور که از نامش نیز مشخص است در صورتی که تعیین شود فایل ویدیویی را به محض بارگذاری و آماده شدن، به صورت خودکار اجرا خواهد کرد.

controls: این صفت بولین در صورتی که تعیین شود مشخص می‌کند که کلید های کنترلی پخش فایل ویدیویی باید نمایش داده شود. مرورگر ها می‌بایست کلید های کنترلی زیر را فراهم کنند. کلید های کنترلی شامل play، volume، seeking، pause، دکمه Fullscreen می‌باشند.

Loop: این صفت بولین در صورت ذکر شدن فایل ویدیویی را پس از پایان مجددا اجرا خواهد کرد.

poster: این صفت موجب نمایش یک عکس در هنگام بارگذاری فایل یا تا زمانی که کاربر فایل را اجرا نکرده است خواهد شد. این صفت آدرس یک عکس را به عنوان مقدار می‌پذیرد. بهتر است این تصویر، تصویر یکی از فریم های فیلم باشد. در صورتی که از این صفت استفاده نشود مرورگر به صورت پیش فرض اولین فریم فیلم را نمایش خواهد داد.

preload: این صفت مشخص می‌کند که آیا فایل ویدیویی باید همزمان با بارگذاری صفحه، دانلود شود یا خیر؟ این صفت سه مقدار زیر را می‌پذیرد:

auto: تعیین این مقدار موجب بارگذاری فایل همزمان با بارگذاری صفحه خواهد شد.

metadata: این مقدار فقط موجب دریافت اطلاعات اضافی فایل و عدم بارگذاری آن همزمان با بارگذاری صفحه خواهد شد. اطلاعات اضافی شامل مواردی همچون ابعاد فیلم، اولین فریم، لیستی از track ها، مدت زمان و... می‌باشد.

none: تعیین این مقدار از بارگذاری فایل همزمان با بارگذاری صفحه جلوگیری خواهد کرد.

src: این صفت هم آدرس فایلی که باید پخش شود را مشخص می‌کند.

width

height

در مثال یک نمونه از کاربرد این تگ همراه با بعضی از صفت های فوق الذکر را مشاهده می‌کنید:

```
<video src="movie.ogg" controls="controls" preload="auto" width="500"
height="400" poster="img_poster.jpg" autoplay="autoplay" audio="muted">
</video>
```


تگ <audio>



از این تگ برای گنجاندن فایل صوتی در صفحات اینترنتی استفاده می‌شود. این تگ از صفت های زیر با همان توضیحات ذکر شده در تگ video پشتیبانی می‌کند:

autoplay controls loop preload src 


یک نمونه کاربرد این تگ را در مثال زیر مشاهده می‌کنید:

```
<audio src="song.ogg" controls="controls">
  Your browser does not support the audio element.
</audio>
```

کد ۱۰-۴

همانطور که می‌دانید همه مرورگرها از تمامی فرمت‌های ویدیویی و صوتی به صورت یکسان پشتیبانی نمی‌کنند. در این صورت چنانچه ما در تگ‌های audio و video با استفاده از صفت src فایلی با فرمت خاص را مشخص کنیم ممکن در بعضی از مرورگرها نمایش داده نشود.

یکی از مزیت‌های تگ‌های فوق امکان تعریف همزمان چندین فرمت از یک فایل برای حل این مشکل است. برای این کار کافی از تگی به نام source استفاده کنیم. ما می‌توانیم همزمان از هر تعداد تگ source برای تعیین فرمت‌های مختلف در داخل تگ‌های audio و video استفاده کنیم. مرورگرها از میان فرمت‌های مشخص شده اولین فرمتی که با آن‌ها سازگار است را اجرا می‌کنند. تگ source در این حالت دو صفت می‌پذیرد که به شرح زیر است:

src: این صفت آدرس فایل مورد نظر را تعیین می‌کند. 

Type: این صفت هم MIME Type فایل مورد نظر را تعیین می‌کند. 

کد زیر نحوه استفاده از تگ‌های audio و source همزمان با یکدیگر را نشان می‌دهد:

```
<audio controls="controls">
  <source src="song.ogg" type="audio/ogg" />
  <source src="song.mp3" type="audio/mpeg" />
  Your browser does not support the audio element.
</audio>
```

کد ۱۰-۵

نحوه استفاده از تگ‌های video و source نیز به همین شکل استفاده است.

فصل یازدهم

ایجاد صفحات خردگانه در HTML

یکی از امکاناتی که HTML برای طراحان فراهم آورده است این است که می‌توانیم به وسیله یکسری عناصر خاص به نمایش چندین صفحه وب در یک پنجره مرورگر بپردازیم. در اصل با این عناصر می‌توانیم پنجره مرورگر را به چندین قسمت (که به هر کدام frame گفته می‌شود) تقسیم کرده و در هر قسمت یک صفحه را نمایش دهیم. در ادامه با ساختار کلی و تگ‌های مربوط به چنین صفحاتی بیشتر آشنا خواهیم شد.

ساختار صفحات چندگانه

ساختار صفحاتی که می‌خواهیم در آن‌ها چندین صفحه را قرار دهیم کمی با صفحه‌های معمولی فرق می‌کند. همانطور که می‌دانید یک صفحه معمولی از قسمت سر که بوسیله تگ `head` و قسمت پیکره که به وسیله تگ `body` مشخص می‌شود تشکیل می‌شود. در زیر ساختار کلی یک صفحه معمولی را مشاهده می‌کنید:

```
<html>
  <head>
    .....
  </head>
  <body>
    .....
  </body>
</html>
```

کد ۱-۱۱

اما در ساختار صفحات دسته دوم به جای عنصر `<body>` از عنصری به نام `<frameset>` استفاده می‌شود. قبل و بعد از این عنصر نباید هیچ عنصر دیگری قرار بگیرد در غیر این صورت `frameset` توسط مرورگرها چشم پوشی می‌شود. ساختار کلی این گونه صفحات به صورت زیر است:

```
<html>
  <head>
    .....
  </head>
  <frameset cols='50%,50%'>
    <frame src='http://www.google.com' name='google_frame' />
    <frame src='http://www.yahoo.com' name='yahoo_frame' />
  </frameset>
</html>
```

کد ۲-۱۱

می‌بینید که در این گونه صفحات از تگ‌های `<frameset>` برای ایجاد یک صفحه چندگانه و تگ `<frame />` برای مشخص کردن صفحاتی که در هر بخش نمایش داده خواهد شد استفاده می‌کنیم. در ادامه به بررسی هر یک از تگ‌های مرتبط با ایجاد صفحات چندگانه آشنا خواهیم پرداخت: قبل از هر چیز اجازه بدهید به این نکته اشاره کنیم که `FRAME`‌ها (همان بخش‌های صفحات چندگانه) می‌توانند به صورت افقی یا عمودی در کنار هم قرار بگیرند.

تگ <FRAMESET>



از این تگ برای ایجاد صفحات چندگانه استفاده می‌کنیم. در واقع با استفاده از این تگ برای اینکه به مرورگر بگوییم با یک صفحه چندگانه روبروست استفاده می‌شود. این تگ صفت‌های زیر را پشتیبانی می‌کند:

rows: برای ایجاد فریم های افقی به کار می رود. از این صفت برای تعیین تعداد و اندازه هر یک از فریم ها استفاده می شود که در واقع لیستی از اندازه ها بر اساس واحد های px، درصد و نسبی خواهد بود.

cols: برای ایجاد فریم های عمودی به کار می رود. از این صفت برای تعیین تعداد و اندازه هر یک از فریم ها استفاده می شود که در واقع لیستی از اندازه ها بر اساس واحد های px، درصد و نسبی خواهد بود.

همانطور که اشاره شد یکی از انواع مقادیری که صفت های rows و cols می پذیرند مقادیر نسبی هستند که با * مشخص می شوند. این مقادیر پس از محاسبه اندازه های مشخص شده با px و درصد بر رویداد فضای باقیمانده عمل می کنند. در مثال زیر ابتدا 300px برای frame اول، سپس 50% از فضای باقیمانده برای frame دوم اختصاص داده می شود. پس از آن فضای باقیمانده به ۶ قسمت کاملاً مساوی تقسیم شده که ۳ قسمت آن برای فریم سوم، دو قسمت برای فریم چهارم و یک قسمت برای فریم پنجم اختصاص داده می شود:

```
<frameset cols="300px,50%,3*,2*,1*">...
```

کد ۱۱-۳

border: ضخامت border بین هر یک از فریم ها را کنترل می کند. عددی صحیح را بر حسب px می پذیرد.

bordercolor: رنگ border اطراف فریم ها را مشخص می کند. به سه روشی که قبلاً به آن ها اشاره شد می توان به این صفت مقدار دهی کرد.

frameborder: نمایش یا عدم نمایش border های اطراف فریم ها را مشخص می کند. مقادیر زیر را می پذیرد:

1 یا **yes:** border ها نمایش داده خواهند شد. ✓

0 یا **no:** border ها نمایش داده نخواهند شد. ✓

framespacing: مثل صفت border رفتار می کند.

تگ <frame />

برای مشخص کردن صفحه ای که در فریم ها نمایش داده می شود از این تگ استفاده می شود. این تگ که جزء تگ های تهی است حتماً باید در داخل تگ <frameset> استفاده شود. دقت کنید تعداد فریم های به کار رفته در داخل frameset ها باید با تعداد اندازه هایی که در صفت rows یا cols تگ frameset آمده است برابر باشد. نه کمتر نه بیشتر !!! این تگ از صفت های زیر پشتیبانی می کند:

name: نامی برای frame مشخص می کند. این نام باید در صفحه یکتا باشد. دقت کنید که این نام می تواند به عنوان پنجره هدف لینک ها مورد استفاده قرار گیرد.

longdesc: آدرسی را که توضیح مفصلی در مورد فریم در آن هست را مشخص می‌کند.

src: آدرس صفحه ای که باید در فریم نمایش داده شود را مشخص می‌کند.

noresize: این صفت بولین در صورتی که برای فریمی به کار رود از امکان تغییر اندازه فریم توسط کاربر جلوگیری خواهد کرد.

scrolling: با این صفت می‌توانیم چگونگی نمایش یا عدم نمایش اسکرول های frame ها را هنگامی که محتویاتشان از اندازه frame بیشتر باشد را کنترل کنیم. این صفت مقادیر زیر را می‌پذیرد:

auto: در صورت لزوم ابزار scroll را برای فریم نمایش خواهد داد.

yes: همیشه ابزار scroll را برای فریم نمایش خواهد داد. (حتی اگر محتویات بیش از اندازه کلی فریم نباشد.)

no: هرگز ابزار scroll را برای فریم نمایش نخواهد داد. (حتی اگر محتویات بیش از اندازه کلی فریم باشد)

frameborder: نمایش یا عدم نمایش border همان فریم را کنترل می‌کند. همان مقادیری را می‌گیرد که صفت Frameborder برای تگ frameset می‌گیرد.

marginwidth: فاصله ی محتویات داخل فریم را از دیواره های چپ و راست (در عرض فریم) را کنترل می‌کند. این صفت عددی بر حسب px یا درصد را می‌پذیرد.

marginheight: فاصله ی محتویات داخل فریم را از دیواره های بالا و پایین (در ارتفاع فریم) را کنترل می‌کند. این صفت عددی بر حسب px یا درصد را می‌پذیرد.

تگ <noframes>

بعضی مرورگر ها کلا از frameset ها (صفحات چندگانه) پشتیبانی نمی‌کنند در نتیجه تگ های frame و frameset را نمی‌شناسند و احتمالاً خود آن ها را در صفحه عینا نمایش خواهند داد. برای جلوگیری از این کار باید از تگی به نام <noframes> در داخل تگ <frameset> برای نمایش یک کد جایگزین در این گونه مرورگر ها استفاده کنیم. این تگ صفت خاصی ندارد و تنها وظیفه آن نمایش کدهایی است که در داخلش قرار می‌گیرد. جالب است بدانید در داخل این تگ شما می‌توانید از انواع تگ ها استفاده کنید. حتی می‌توانید ساختار اصلی یک صفحه وب معمولی را در آن قرار داده تا به جای frameset ها نمایش داده شوند.

تگ <iframe>

تگ iframe (inline Frame) معمولاً برای قرار دادن یک صفحه در داخل صفحه ای معمولی و در کنار سایر عناصر مانند متن ها، عکس ها، لینک ها و... استفاده می‌شود.

این تگ از تمام صفات تگ `<frame>` که قبلا به آن اشاره شده است پشتیبانی می‌کند به غیر از `noresize`. اما از صفت های خاصی به نام `width` و `height` که برای تعیین عرض و ارتفاع محدوده `frame` استفاده می‌شود پشتیبانی می‌کند. در یک کلام قرار دادن یک صفحه داخل صفحه دیگر توسط تگ `<iframe>` همانند قرار دادن یک عنصر `<object>` در صفحه است.

فصل دوازدهم

قرار دادن سایت بر روی وب

موقعی که سایتتان را ایجاد کردید ، مطمئناً می‌خواهید آن را برای سایر افراد قابل مشاهده کنید. در این فصل با نحوه آماده سازی سایت برای نمایش عموم و قرار دادن آن بر روی وب آشنا خواهیم شد. پس از آماده شدن سایت برای انتشار عمومی آن ، نیاز به بررسی دقیق انواع هاستینگ خواهیم داشت. ما در این فصل شما را با اصطلاحات کلیدی استفاده شده توسط شرکت های میزبانی وب همچون فضا ، پهنای باند و... به منظور کمک در انتخاب صحیح بسته های ارائه شده از طرف آن ها ، آشنا خواهیم کرد.

اما قبل از همه این ها نیاز به بررسی آخرین عنصر یعنی `<meta />` که وظیفه فراهم آوردن اطلاعاتی در مورد صفحات و محتوای آن ها را بر عهده دارد خواهیم داشت که این عنصر را نیز به صورت دقیق با تمام جزئیات بررسی خواهیم کرد.

تگ <meta />



از این تگ تھی که باید همیشه در قسمت head قرار بگیرد برای فراهم کردن یکسری اطلاعات در مورد محتوای صفحه استفاده می‌شود. مثلا اینکه نویسنده صفحه چه کسی بوده، تکنولوژیهای به کار رفته در آن، تاریخ ایجاد و انقضای صفحه، برنامه تولید کننده صفحه و.... این اطلاعات می‌توانند توسط مرورگرها برای تشخیص نحوه نمایش محتوا یا توسط موتورهای جستجو برای ذخیره کردن اطلاعات صفحه در پایگاه داده هایشان مورد استفاده قرار بگیرد. به عنوان مثال با تگ meta می‌توانیم یکسری کلمات کلیدی مرتبط با صفحه را به موتورهای جستجو معرفی کنیم.

ما از هر تعداد تگ meta در قسمت head می‌توانیم استفاده کنیم. هر تگ meta یک خاصیت و مقدار آن خاصیت را مشخص کند. این تگ از صفت‌های زیر پشتیبانی می‌کند:



name: مقدار این صفت در واقع نام خاصیتی است که می‌خواهیم آن را تنظیم کنیم. مقداری که به این صفت داده می‌شود می‌تواند هر چیزی باشد. اما بعضی از مقادیر از پیش تعیین شده و پر استفاده برای این صفت به شرح زیر می‌باشند.

description: این مقدار مشخص می‌کند که می‌خواهیم توضیحاتی در مورد صفحه را تعیین نماییم.

keywords: از این مقدار برای مشخص کردن لیستی از کلمات کلیدی صفحه استفاده می‌شود. کلمات کلیدی به کلماتی گفته می‌شود که انتظار داریم کاربران با جستجوی آن‌ها در موتورهای جستجو به سایت ما هدایت شوند.

robots: این مقدار چگونگی ذخیره کردن صفحه توسط برنامه‌های موتورهای جستجو را مشخص می‌کند. معمولا از این مقدار برای جلوگیری از ذخیره شدن صفحه در موتورها یا جلوگیری از دنبال شدن لینک‌ها توسط موتورهای جستجو استفاده می‌شود.



content: این صفت که صفتی الزامی برای تگ meta نیز هست ارتباط نزدیکی با صفت name دارد. در واقع این صفت حاوی اطلاعاتی در باره خاصیتی است که در صفت name، نام آن ذکر شده است. در مثال‌های زیر چند نمونه از کاربرد تگ meta همراه با این دو صفت را مشاهده می‌کنید:

```
<meta name="keywords" content="HTML,XHTML,Learning,download,Ebook">
```

کد ۱-۱۲

این دستور یکسری کلمات کلیدی برای صفحه را مشخص می‌کند. کلمات کلیدی می‌بایست حداکثر ۲۰۰ کارکتر باشد.

```
<meta name="description" content="best reference for web design and learning HTML and XHTML">
```

کد ۲-۱۲

این دستور یک توضیح کلی در مورد محتوای صفحه را مشخص می‌کند.

به این نکته نیز توجه داشته باشید که محتوای description حداکثر باید ۲۰۰ کاراکتر باشد هر چند موتور جستجوی گوگل فقط ۱۰۰ کاراکتر ابتدایی را در نتایج جستجو نشان می‌دهد.
صفت content در حالتی که مقدار صفت name برابر robots تعیین شده است می‌تواند مقادیر زیر را بپذیرد:

all: ذخیره تمامی صفحات

none: ذخیره هیچ کدام از صفحه‌ها

index: فقط ذخیره صفحه کنونی

noindex: عدم ذخیره صفحه کنونی

follow: دنبال کردن تمامی لینک‌های موجود در صفحه

nofollow: عدم دنبال کردن لینک‌های موجود در صفحه

به عنوان مثال در کد زیر هیچ موتوری نباید صفحه جاری را ذخیره و لینک‌های موجود در آن را دنبال کند:

```
<meta name="robots" content="noindex,nofollow" />
```

کد ۱۲-۳

http-equiv: این صفت به صورت اختصاصی اطلاعاتی از صفحه را در اختیار سرورها قرار می‌دهد. مثلاً یکی از کاربرد های این صفت برای مشخص کردن نوع encoding صفحه برای سرور ها (و حتی مرورگر هاست). این صفت مقادیر زیادی را می‌پذیرد که پرکاربرد ترین آن‌ها به شرح زیر است:

content-type: برای مشخص کردن نوع Mime Type صفحه که غالباً text/html است و نوع encoding صفحه استفاده می‌شود. به عنوان مثال دستور زیر نوع encoding صفحه را برابر UTF-8 که غالباً برای صفحات فارسی هم استفاده می‌شود قرار می‌دهد:

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

کد ۱۲-۴

refresh: با استفاده از این مقدار می‌توان مشخص کرد که صفحه بعد از مدت زمانی معین خود به خود بارگذاری مجدد شود. دستور زیر صفحه را بعد از ۵ ثانیه refresh خواهد کرد:

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

کد ۱۲-۵

از این مقدار همچنین برای انتقال به صفحات دیگر نیز استفاده می‌شود. مثلاً در مواقعی که آدرس سایت شما عوض شده است و می‌خواهید کاربران را از آدرس قبلی به آدرس جدید به صورت خودکار منتقل کنید می‌توانید از مقدار refresh به روش زیر استفاده کنید:

```
<meta http-equiv="Refresh" content="5;URL=http://www.your-news-site.com" />
```

کد ۱۲-۶

در کد فوق کاربر پس از ورود به صفحه بعد از ۵ ثانیه به صورت خودکار به آدرس www.your-new-site.com منتقل خواهد شد.

هنگامی که از این تکنیک برای بارگذاری مجدد یک صفحه استفاده می‌کنید به آن refreshing و زمانی که از آن برای انتقال خودکار از یک صفحه به صفحه دیگر استفاده می‌کنید به آن Redirecting اطلاق می‌شود.

expires ^v از این مقدار برای تعیین زمان انقضای صفحات استفاده می‌شود. در مثال زیر مرورگر مجبور است بعد از تاریخ ذکر شده به جای فراخوانی صفحه از **cache** (حافظه نهان مرورگر) یک کپی جدید آن را از سرور درخواست کند:

```
<meta http-equiv="expires" content="Fri,16 September 2011 23:59:59 pm" />
```

کد ۱۲-۷

pragma ^v با تعیین این مقدار برای صفت http-equiv می‌توان از ذخیره شده صفحه در حافظه نهان مرورگر جلوگیری کرد. در این حالت می‌بایست صفت content برابر عبارت no-cache باشد:

```
<meta http-equiv="pragma" content="no-cache" />
```

کد ۱۲-۸

author ^v از این مقدار برای تعیین نام طراح و نویسنده صفحه استفاده می‌شود:

```
<meta http-equiv="author" content="ali Hosseini" />
```

کد ۱۲-۹

generation ^v از این مقدار نیز برای مشخص کردن نام برنامه استفاده شده برای طراحی صفحه ، استفاده می‌شود:

```
<meta http-equiv="generation" content="Adobe DreamWeaver" />
```

کد ۱۲-۱۰

هنگامی که از صفت http-equiv استفاده می‌کنید نباید از صفت name استفاده کنیم. عکس این حالت نیز باید رعایت شود.

گرفتن نام دامنه

دامنه به آن بخش از آدرس اینترنتی وب که معمولاً پس از حروف www می‌آید گفته می‌شود. به عنوان مثال گوگل از نام دامنه google.com استفاده می‌کند در حالی که سایت آمازون از دامنه amazon.com در کشور آمریکا و از دامنه amazon.de در کشور آلمان استفاده می‌کند.

شما می‌توانید برای ثبت دامنه مورد نظرتان به هزاران سایت شرکت های ثبت کننده دامنه موجود در اینترنت مراجعه کنید. بسیاری از این شرکت ها پسوند.com را به شما پیشنهاد می‌کنند در حالی که پسوند های دیگری نیز قابل ثبت است. به عنوان مثال پسوند های مختص کشور ها همچون.uk برای کشور اوکراین،.de برای کشور آلمان،.ir برای ایران و... وجود دارد. از پسوندهای مختص کشور ها فقط وقتی باید استفاده کنید که در آن کشور ها مستقرید. البته پسوند های دیگری که هر کدام نشان دهنده موضوع خاصی هستند نیز وجود دارد. مانند.info برای سایت های مبتنی بر اطلاعات،.org برای سایت های متعلق به ارگان ها،.ac.ir برای سایت های دانشگاهی ایران.

قبل از ثبت نام دامنه مورد نظرتان می‌بایست آزاد بودن آن را بررسی کنید. تمامی شرکت های ثبت دامنه می‌بایست فرمی برای بررسی آزاد بودن دامنه مورد نظرتان در اختیار قرار دهند. در ادامه به بررسی ویژگی های یک دامنه خوب اشاره خواهیم کرد:

کوتاه بودن دامنه

کم تعداد بودن حروف یک دامنه از چند لحاظ مورد اهمیت است. دامنه های کوتاه بیاد ماندنی هستند و خود باعث تبلیغ دهان به دهان آدرس وب سایت شما می‌شود و البته سریع هم تایپ می‌شوند. از انتخاب دامنه های طولانی و آن هایی که یادآوریشان دشوار است پرهیز کنید. به عنوان مثال شما می‌توانید به جای انتخاب دامنه ای همچون www.payamnouruniversity.com از نامی به صورت www.pnu.com استفاده کنید. ضمناً موتور های جستجو به دامنه های کوتاه تر بیشتر اهمیت می‌دهند.

مرتبط بودن با محتوا

بهتر است نام دامنه انتخابی مرتبط با نوع فعالیت سایت، مطالب محتوایی آن و یا شرکتهای باشد. این گونه دامنه ها معمولاً خوب در ذهن مانده و به راحتی یاد آوری می‌شوند. علاوه بر این در نگاه اول معرف شما نیز خواهند بود.

شامل کلمات کلیدی باشد

سعی کنید از پر استفاده ترین کلمات کلیدی سایتتان در نام دامنه ای که انتخاب می‌کنید استفاده شود. این کار علاوه بر معرفی کردن موضوع سایت به کاربران، موجب نمایش سایت شما در رتبه های بالای جستجو های مرتبط با آن کلمات کلیدی در موتور های جستجو خواهد شد.

میزبانی وب^۱

برای قرار دادن سایتتان بر روی وب نیاز به مقدار فضایی از وب سرور ها خواهید داشت. هزاران شرکت امکان قرار دادن سایتتان را بر روی سرور هایشان می‌دهند. از آنجایی که این شرکت ها میزبانی سایت شما را بر عهده دارند به آن ها web hosting نیز می‌گویند.

اصطلاحات کلیدی در انتخاب یک میزبان

در این قسمت به بررسی نکات کلیدی که شما نیاز به فهمیدن و پیروی از آن ها در هنگام انتخاب یک میزبان خوب خواهید داشت می‌پردازیم:

پشتیبان گیری^۲

شما باید در مورد اینکه آیا میزبان مورد نظرتان از سایتتان پشتیبان گیری می‌کند یا نه و در مورد چگونگی انجام آن تحقیق کنید.

منظور از پشتیبان گیری یک کپی ساده از محتویات هر سایت به منظور بازگردانی آن ها در صورت بروز مشکل در کامپیوتر هایی است که سایت شما بر روی آن ها قرار دارد.

پهنای باند^۳

به مقدار داده هایی که شما مجاز به استفاده از آن ها هستید گفته می‌شود. به عنوان مثال اگر میانگین حجمی صفحات وب شما (شامل تصاویر) 75KB باشد و ماهانه ۱۰۰ بازدید کننده که هر کدام از آن ها ۱۰ بازدید از صفحات داشته باشند آنگاه شما نیاز به حداقل 75000KB (یا 75MB) پهنای باند خواهید.

کشور

می‌بایست در مورد کشوری که سایتتان در آن میزبانی می‌شود اطلاع داشته باشید. بهتر است هاست شما در همان کشوری که هستید قرار داشته باشد در این صورت طی کردن فاصله کمتر داده ها تا رسیدن به مقصد ، موجب نمایش سریعتر سایت به کاربران خواهد شد.

فضای دیسک^۴

به مقدار فضایی از کامپیوتر سرویس دهنده که برای ذخیره فایل هایتان گرفته اید گفته می‌شود. می‌بایست همیشه فضایی که خریداری می‌کنید از مجموع حجم تمامی فایل های بر روی سایتتان همچون فایل های HTML ، CSS ، javascript ، تصاویر استفاده شده در صفحات و... کمی بیشتر باشد.

¹ hosting

² backups

³ Bandwidth

⁴ Disk space

مراکز داده^۱

به مکان هایی که کامپیوتر های سرویس دهنده شرکت های میزبانی وب در آن ها قرار دارند گفته می شود.

Uptime

به درصد زمانی که کامپیوتر سرویس دهنده سایت شما ، می تواند کار کند و برای بینندگان سایتان قابل دسترس باشد گفته باشد. به عنوان مثال آپ تایم ۹۹ درصد به معنی این است که ۹۹ دقیقه از هر ۱۰۰ دقیقه سایت شما در دسترس خواهد بود و ممکن است در ۱ درصد از مواقع که به معنی ۸۷.۶ ساعت یا حدود ۴ روز در سال است از دسترس خارج باشد.

حساب های کاربری ایمیل^۲

عموما شرکت های هاستینگ علاوه بر خدمات میزبانی وب به ارائه خدمات ایمیل نیز اقدام می کنند. شما در مورد می بایست به دو فاکتور مهم توجه کنید: حجم قابل استفاده برای هر یک از حساب های ایمیل و دیگری تعداد حساب های ایمیلی که شما دریافت می کنید.

قرار دادن سایت بر روی سرور از طریق FTP

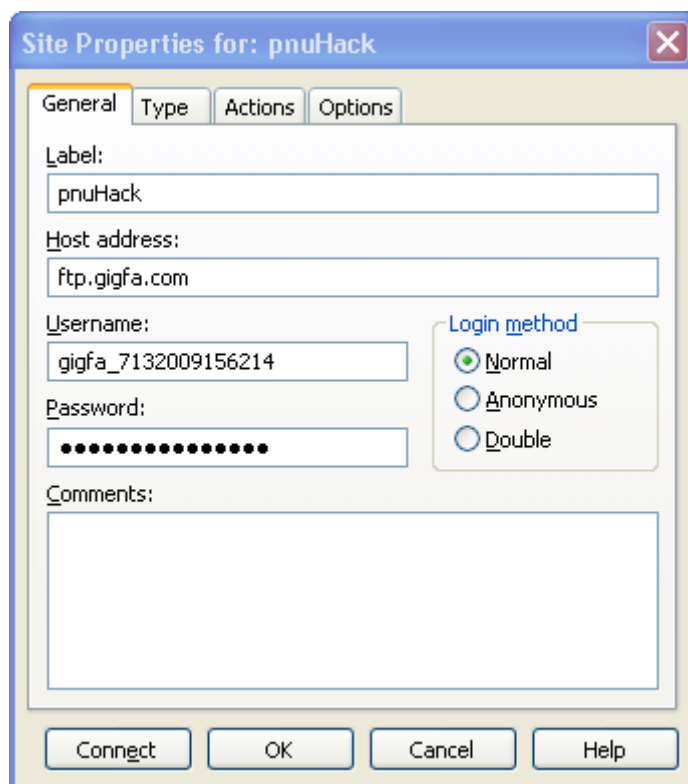
پس از آنکه مقدار فضایی از یک کامپیوتر سرویس دهنده را خریداری کردید ، نیاز به قرار دادن فایل های تشکیل دهنده سایتان بر روی آن خواهید بود. بهترین راه برای انجام این کار استفاده از FTP است. FTP یکی از انواع پروتکل های موجود در اینترنت است که برای ارسال داده ها در داخل اینترنت استفاده می شود و بسیار سریعتر از پروتکلی همچون HTTP در ارسال وبسایت ها بر رویداد سرور ها عمل می کند. بسیاری از شرکت های میزبانی وب شما را مجبور استفاده از این پروتکل برای انتقال صفحاتتان بر روی سرورهایشان می کند که این به معنی این است که شما به یک برنامه FTP نیاز خواهید داشت. برنامه های FTP اغلب از دو پنجره که هر کدام دارای یک جستجوگر فایل هستند تشکیل می شود: یکی که نمایش دهنده فایل ها و پوشه های موجود بر روی کامپیوتر خودتان و دیگری نمایش دهنده پوشه های موجود بر روی وب سرور است. از جمله برنامه های FTP می توان به FireFtp و CuteFtp برای سیستم عامل های ویندوز و مکینتاش ، FTPX برای ویندوز و Fetch و Transmit برای مکینتاش اشاره کرد. پس از خریداری و ثبت هاست ، شرکت هاستینگ اطلاعات لازم برای انتقال فایل ها از طریق FTP که شامل موارد زیر است را برای شما ارسال می کند:

- آدرس FTP (مانند ftp.example.com)
- شناسه کاربری FTP
- کلمه عبور FTP

در ادامه به نحوه کار با برنامه CuteFtp برای انتقال فایل ها خواهیم پرداخت.

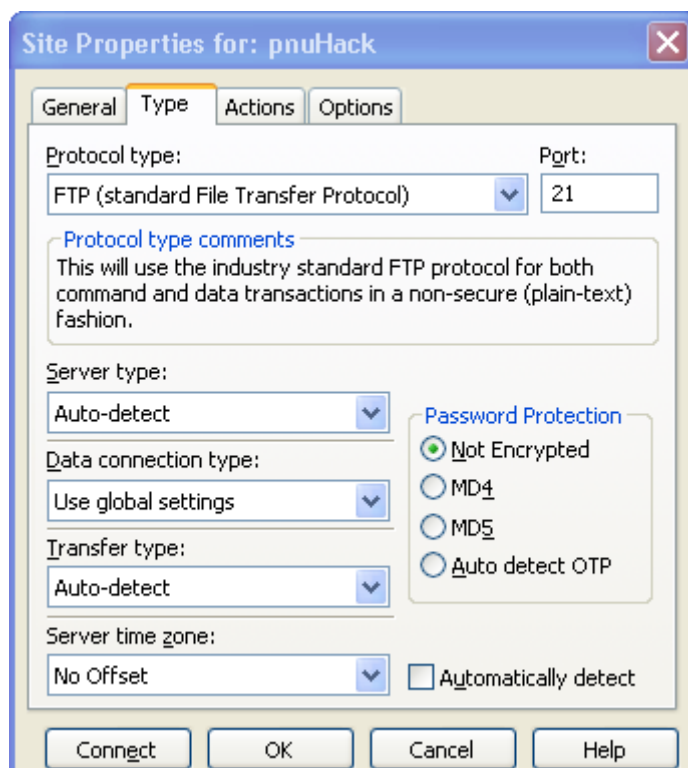
کار با برنامه CuteFtp

قبل از هر چیز می‌بایست برای انتقال فایل‌تان بر روی سرور یک اتصال (connection) جدید ایجاد کنید. برای این کار کافی است از منوی File گزینه New و سپس گزینه Ftp Site را انتخاب کنید. حال در پنجره باز شده و در قسمت‌های مربوطه مشخصات ارائه شده از طرف هاست را وارد نمایید:




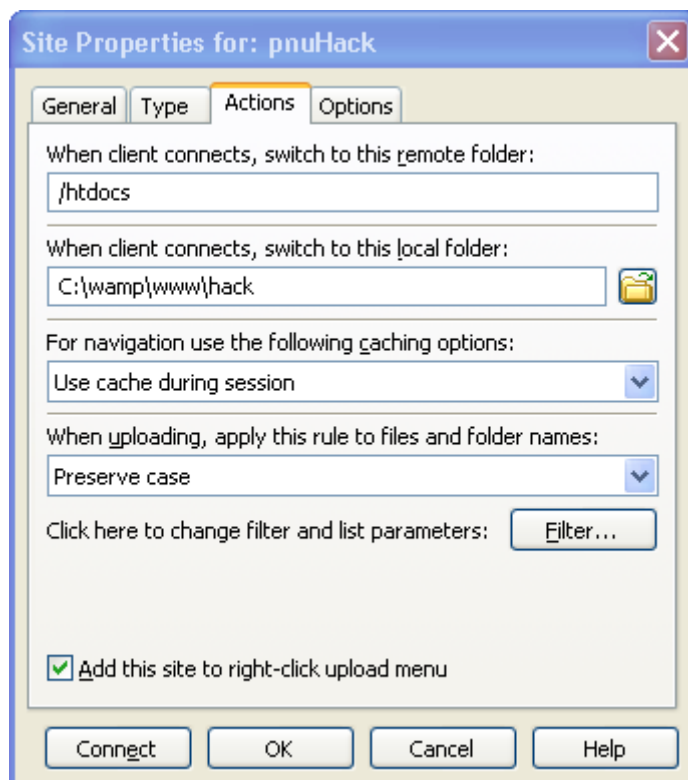
شکل ۱-۱۲

حال در سربرگ type مقدار Type protocol را برابر FTP(standard file transfer protocol) قرار دهید. همچنین می‌بایست گزینه‌های server Type و Transfer Type را برابر Auto-detect قرارداده و سایر گزینه‌ها را در حالت پیش فرض رها کنید:



شکل ۱۲-۲

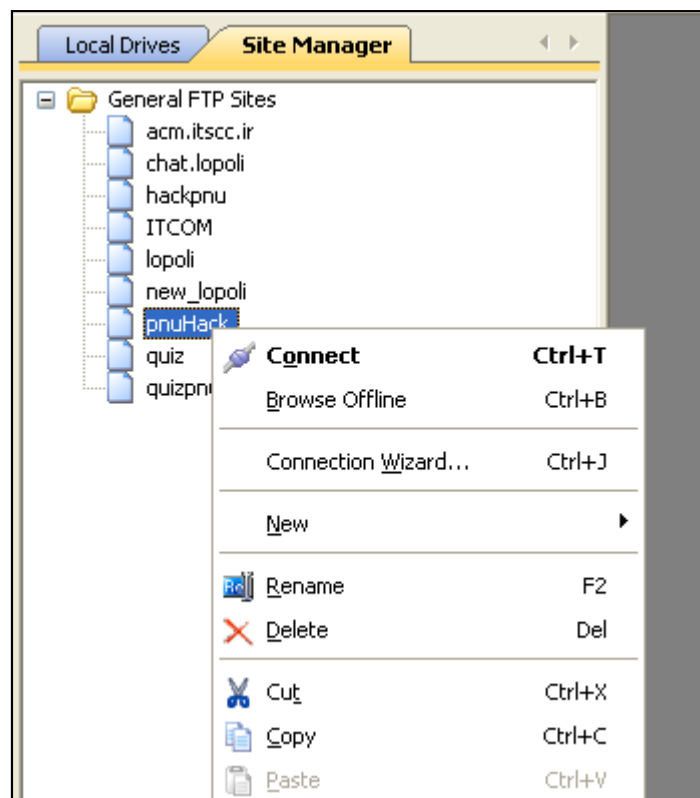
در سربرگ Actions در فیلد ابتدایی ، آدرس پوشه ای از داخل هاست که می‌خواهید در هر بار اتصال به صورت پیش فرض باز شود را وارد کنید. این مسیر اغلب به صورت public_html یا WWW است. در فیلد دوم هم می‌توانید با کلیک بر روی تصویر  مسیر پوشه ای از کامپیوتر خودتان که حاوی فایل های سایتتان می‌باشد را انتخاب کنید:



شکل ۱۲-۳

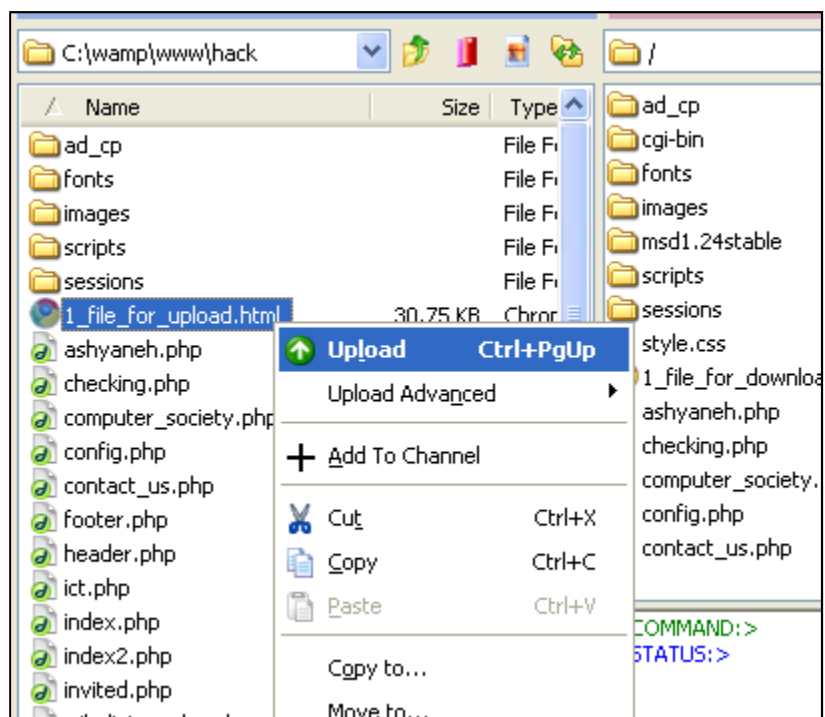
در انتها بر روی دکمه OK کلیک کنید تا اتصال جدید شما به لیست اتصالات موجود در قسمت Site Manager اضافه شود.

حال برای برقراری اتصال به منظور ارسال فایل ها می‌توانید بر روی اتصال مورد نظرتان کلیک راست کرده از منوی باز شده گزینه Connect را انتخاب کنید:



شکل ۱۲-۴

پس از برقراری اتصال فایل های موجود در کامپیوتر شما و سرور دهنده نمایش داده خواهد شد. اکنون برای انتقال یک فایل از کامپیوترتان به سرور کافی است پس از کلیک راست بر روی فایل مورد نظرتان گزینه Upload را انتخاب کنید:



شکل ۱۲-۵

پس از آن عملیات آپلود شروع شده و نحوه انجام و درصد پیشرفت آن در قسمت Queue Window نمایش داده می‌شود.

ضمیمه

تفاوت های HTML و XHTML

زبان XHTML که مخفف EXtensible HyperText Markup Language در تاریخ ۲۶ ژانویه سال ۲۰۰۰ منتشر شد و به قول بعضی ها همان HTML 5 می باشد که در واقع اینچنین نیست. این زبان نسل جدید زبان وب می باشد که دارای یکسری اصول و قوانین خاص است تا کمبودهای HTML را جبران کند. در ادامه به برخی تفاوت های این نسخه با نسخه قبل آن خواهیم پرداخت که شما باید در کد نویسی هایتان از آن ها پیروی کنید:

- نوشتن تگهای </head></body> که در HTML اجباری نبود در XHTML اجباری است.
- وجود تگهای پایان دهنده اجباری است مانند:
, <hr />, <p>...</p>.
- کلیه تگها و خصوصیات آنها باید بصورت حروف کوچک انگلیسی نوشته شود. مانند:

```
<body bgcolor="#ffcc45">
```

کد ض- ۱

- مقادیر خصوصیات در تگها باید بین دو علامت "..." قرار گیرند.
- ترتیب قرار گرفتن تگها داخل هم باید رعایت شود.

```
<b><i> TEXT </b></i>    روش غلط  
<b><i> TEXT </i></b>    روش صحیح
```

کد ض- ۲

- بعضی از خصوصیات در HTML نیاز به مقداردهی نداشتند اما در XHTML آنها را باید مقداردهی کنیم،

```
<input type="radio" checked>    روش غلط  
<input type="radio" checked="checked"> xhtml    روش صحیح در
```

کد ض- ۳

- در تگ <pre> فقط از متن باید استفاده کنید و داخل آن از تگهای دیگر نمی توانید استفاده کنید.
- یک فرم را داخل فرم دیگر نباید بکار برید.
- دستورات فایل CSS باید به صورت حروف کوچک نوشته شوند و بهتر است که برای Style sheet از فایل خارجی استفاده کنید.

MIME TYPE چیست ؟

یک مرورگر و در حالت کلی سرور ها و کلاینت ها هیچ گاه نوع فایل هایی را که برای همدیگر ارسال می کنند از روی پسوند آن ها تشخیص نداده و به جای آن از مفهومی به نام MIME Type برای این کار استفاده می کنند. اگر از شما سوال شود که مثلا نوع فایلی به نام index.htm چیست احتمالا می گوئید خوب HTML ! در حالی که راه های زیادی برای تغییر پسوند انواع فایل ها به غیر از آن چیزی که واقعا هستند وجود دارد. پس اگر مرورگر ها بخواهند برای تشخیص نوع فایل ها از این راه استفاده کنند قطعا دچار مشکل خواهند شد.

در واقع مرورگر ها برای تشخیص نوع (طبیعت) یک فایل از Mime Type آن استفاده می کنند نه از پسوندی که از طریق URL برایشان ارسال می شود.

یکی از قوانینی که در پروتکل HTTP همه کامپیوتر ها ملزم به رعایت آن هستند این است که Mime Type فایل هایی که برای یکدیگر ارسال می کنند را نیز بفرستند.

Mime Type ها ساختار بسیار ساده ای دارند که در حالت کلی به شکل زیر است:

Type/subtype

فایل های مختلف Mime Type های مختلف و البته منحصر به فرد خود را دارند. به عنوان مثال در زیر Mime Type برخی از انواع فایل ها که استفاده زیادی در صفحات وب دارند را مشاهده می نمایید:

نوع فایل	Mime Type
.gif	image/gif
.jpg	image/jpeg
.png	image/png
.html or .htm	text/html
.css	text/css
.js	text/javascript
.pdf	application/pdf
.swf	application/x-shockwave-flash
.zip	application/zip
.mp3	audio/mpeg

کد های زبانی^۱

بعضی از صفت های تگ ها در HTML مقداری را به عنوان کد زبانی می پذیرند. مثلا صفت lang در بسیاری از تگ ها مانند p وجود دارد که یک کد زبانی را برای مشخص کردن اینکه متن داخل آن به چه زبانی نوشته شده است می پذیرد. کد های زبانی معمولا دو حرفی هستند و برای هر زبان موجود در دنیا کد زبانی یکتا تعریف شده است مثلا: fa (برای فارسی)، en (برای انگلیسی)، fr (برای فرانسوی) و ...

اما بعضی از کد های زبانی حالت زیر مجموعه ای داشته مثلا برای زبان انگلیسی آمریکایی کد زبانی en-US و برای زبان انگلیسی بریتانیایی کد زبانی en-GB را تعریف کرده اند. در حالت کلی کد های زبانی زیر مجموعه ای به شکل زیر هستند:

¹ language code

Language code: primary code("-"subcode)

کدهای کاراکتری^۱

همانطور که می‌دانید بعضی از کاراکترها در زبان HTML کار خاصی را انجام می‌دهند و در واقع مفهوم خاصی دارند. مثلا علامت < (less than) به معنای شروع یک تگ و علامت > (greater than) به معنای پایان یک تگ است. به این کاراکترها، **کاراکترهای ویژه** می‌گویند.

گاهی اوقات لازم است خود این کاراکترها را مستقیما در صفحه چاپ کنیم. طبیعتا اگر بخواهیم آنها را مستقیما در source صفحه وارد کنیم مرورگرها در مواجهه با آنها، این کاراکترها را به عنوان شروع یا پایان تگ در نظر می‌گیرند و این احتمالا موجب به هم ریختگی صفحه و نتایج غیرمنتظره خواهد شد!

برای حل این مشکل می‌بایست از کدهای کاراکتری معادل آنها استفاده کنیم.

کدهای کاراکتری با & شروع می‌شوند و به ; ختم می‌شوند و در بین این دو علامت هم عبارت نشان دهنده کاراکتر خاص قرار می‌گیرد. به عنوان مثال در جدول زیر چندین کد کاراکتری معروف و مفید را مشاهده می‌کنید:

کاراکتر	توضیحات	کد اسمی کاراکتری	کد عددی کاراکتری
	non-breaking space	 	
<	less than	<	<
>	greater than	>	>
&	ampersand	&	&
©	copyright	©	©
®	registered trademark	®	®

کدهای کاراکتری دو نوع هستند. کدهای کاراکتری همچون < که از مخفف کاراکتر اصلی استفاده می‌کنند **کدهای اسمی** نامیده می‌شوند.

ما برای نمایش کاراکترهای ویژه از **کدهای عددی** نیز می‌توانیم استفاده کنیم. یک کد عددی از یک علامت &، یک علامت #، عددی ویژه که نشان دهنده کاراکتر است و یک علامت ; تشکیل شده است. در جدول بالا نمونه ای از این کدها و کاراکتر معادلشان را مشاهده می‌کنید.

¹ Code entities