

نکاتی پیرامون HTML 4

مبنای این دروس نسخه ۴ استاندارد HTML بوده و نکته بسیار مهم این نسخه جداسازی فرمت‌بندی و انتقال آن به style sheet ها است.

در حالت کلی اجزای یک صفحه وب را می‌توان به سه لایه تقسیم کرد:

۱. لایه محتوا (Content): مشخص کردن محتوا (و محل) قرارگیری آنها در صفحه وب. حاوی اطلاعاتی که قصد نمایش آنها را به کاربر داریم ← HTML
۲. لایه نمایش (Presentation): چگونگی نمایش محتوا را مشخص می‌کند ← CSS
۳. لایه رفتار (Behavior): واکنش به رویدادهای کاربر ← Java Script

یک فایل HTML چیست؟

- HTML برگرفته از حروف اول Hyper Text Markup Language است.
- یک فایل HTML فایلی از نوع text است که متشکل از markup tag می‌باشد.
- مرورگر یا Browser از روی markup tag می‌فهمد که چگونه باید صفحه را نمایش دهد.
- یک فایل HTML باید دارای پسوند htm و یا html است.
- از آنجایی که یک فایل HTML فایلی از نوع text می‌باشد، با هر ادیتور ساده‌ای قابل ایجاد است.
- وظیفه اصلی تگ‌های HTML (markup tags) بیان چگونگی نمایش اطلاعات است.

تگ‌های HTML (HTML Tags)

- با کمک تگ‌های HTML عناصر و یا المان‌ها (Elements) ساخته می‌شوند.
- در زبان HTML حدود ۸۰ عنصر تعریف شده است.
- تگ‌های HTML بوسیله دو کاراکتر < و > ساخته می‌شوند.
- تگ‌های HTML معمولاً بصورت زوج ظاهر می‌شوند، مانند test
- تگ اول در یک زوج تگ مثلاً تگ شروع و تگ دوم مثلاً تگ پایانی نام دارد.
- متن بین تگ اول و تگ دوم در یک زوج تگ محتوای عنصر یا element نامیده می‌شود.
- تگ‌های HTML case sensitive نیستند. برای مثال دو تگ و معادل هم هستند ولی توصیه می‌شود که به خاطر سازگاری با XHTML از حروف کوچک استفاده شود.

توجه: محتویات و نوشته‌های مربوط به هر برچسب باید در بین برچسب‌های شروع و پایان آن برچسب قرار گیرد. [<TITLE> </TITLE>] همچنین برچسبها باید به صورت تو در تو (Nested Tags) باشند نه اینکه در نقاطی با هم همپوشانی (Overlap) داشته باشند پس <X><Y></X></Y> نادرست و <X><Y></Y></X> درست است.

شناسه‌های یک تگ (Tag Attributes)

تگ‌ها می‌توانند حاوی اطلاعات اضافی دیگر باشند، به این اطلاعات شناسه یا Attribute می‌گویند و وظیفه آنها بیان دیگر اطلاعات یک عنصر یا Element است. مثلاً در مورد تگ <body> شناسه‌ای به نام bgcolor وجود دارد که رنگ زمینه متن (background) را تعیین می‌کند. شناسه‌ها به صورت کلی "مقدار=نام" یا "name=value" نوشته می‌شوند و همیشه به تگ شروع یک عنصر یا Element اضافه می‌شوند و نهایتاً اگر در یک عنصر یا Element شناسه‌ها قید نشوند از مقادیر پیش فرض یا default مرورگر وب استفاده خواهد شد.

برخی نکات مهم:

- `<html>` از روی این تگ، مرورگر نوع متن یعنی HTML بودنش را یافته و از روی `</html>` انتهای متن HTML را پیدا می‌کند.
 - متن بین تگ `<head>` و تگ `</head>` اطلاعات شناسنامه ای یا "information Header" متن HTML بوده و نمایش داده نخواهند شد. تگ `head` برای کارهایی از قبیل تعریف کردن عنوان صفحه (Title)، اطلاعات مورد نیاز برای موتورهای جستجو، نوشتن اسکریپتها و... به کار می‌رود.
 - متن بین تگ `<body>` و انتهای آن یعنی تگ `</body>` تنها اطلاعاتی است که توسط مرورگر نمایش داده خواهند شد.
 - مرورگر بیش از یک فاصله خالی (Space) را تشخیص نمی‌دهد. یک کد کوچک و جالب وجود دارد که به معنای فاصله خالی است: ` `
 - در مورد برچسب‌های `<P>` و `
` شما نمی‌توانید بیش از یکبار از آنها استفاده کنید مثلاً استفاده از `<P><P>` موجب رد شدن ۲ خط خالی نمی‌شود و همانند `<P>` عمل می‌کند. حالا برای ایجاد چند خط خالی چه باید کرد؟
 - در نسخه های ۳,۲ و قبلی‌تر زبان HTML از تگ `` برای تعیین نام قلم، رنگ و اندازه متون استفاده می‌شود. در این تگ شناسه `face` نام قلم، شناسه `size` اندازه قلم و شناسه `color` معرف رنگ متن خواهد بود.
 - تگ `` از رده خارج محسوب می‌شود و کنسرسیوم جهانی تعیین استانداردهای وب (World Wide Web Consortium, W3C) این تگ را از مجموعه تگ‌های استاندارد HTML کنار گذاشته است و اکیدا توصیه می‌شود که از `Style Sheet`ها برای تعریف هر گونه خواص نمایشی و آرایشی (layout & Presentation) المانها در HTML استفاده شود.
 - در کل می‌توانید شانس خود را با مشخص کردن بیش از یک فونت بالا ببرید.
- ```

```
- با توجه به توضیحات ارائه شده پیرامون وظایف HTML و CSS، استفاده از تگ‌هایی همانند `font a, i, b` و... به این دلیل که در نحوه‌ی نمایش دخالت می‌کنند مجاز نمی‌باشد. از طرفی تگ‌هایی مانند `em, strong` و... با اینکه مرورگرها به طور پیش فرض نحوه نمایش آنها را تغییر می‌دهند، اما از این جهت استفاده از آنها دارای اهمیت است که می‌تواند اهمیت، تاکید و... یک متن را مشخص کند و از امر می‌تواند برای موتورهای جستجو اهمیت داشته باشد.
  - اگرچه روش‌های متفاوتی برای نمایش متن در صفحه وجود دارد، اما استفاده از تگ‌های `UL, H, P` و... می‌تواند به لحاظ معنایی میزان اهمیت و کیفیت متن را مشخص کند و در مقوله موتورهای جستجو و وب معنایی این مورد دارای اهمیت است.

## رنگ در وب:

همانطور که می‌دانید در کامپیوتر رنگ با کمیت RGB مشخص می‌شود (آبی، سبز، قرمز) که هر کدام از رنگ‌ها می‌تواند عددی بین ۰ تا ۲۵۵ را اختیار کند در فایل‌های HTML هم از همین استاندارد پیروی می‌شود البته اعداد به صورت هگزا دسیمال بیان می‌شوند. روش مشخص کردن رنگ‌ها:

- `rgb(35%, 0%, 50%)`
- `rgb(89, 0, 127)`
- `#59007F`
- `black`

## تصاویر:

- در صفحات HTML از تصاویر با فرمت‌های متفاوتی همانند jpg و gif می‌توان استفاده کرد.
- تصاویر با فرمت gif می‌توانند ساده و یا gif animation باشند ولی در نامگذاری هر دو دارای انشعاب یا extension ی با نام gif هستند.
- هر تصویری با فرمت gif و یا jpg قابل استفاده به عنوان تصویر زمینه می‌باشد. اگر ابعاد تصویر کوچکتر از ابعاد صفحه مرورگر باشد، تصویر زمینه تکثیر خواهد شد.
- توجه داشته باشید که در خصوصیت src باید مسیر عکس مشخص شود نه صرفاً نام آن. در دستورات بالا که ما مسیری را معین نکردیم باید در همان مسیری که فایل HTML ما وجود دارد یک عکس با نام logo و پسوند gif. وجود داشته باشد. بدیهی است که اگر در این مسیر عکسی با این مشخصات وجود نداشته باشد مرورگر عکسی نمایش نخواهد داد!
- به مثال‌های زیر توجه کنید:

```
src="images/logo.gif"
```

یعنی در همان مسیری که فایل HTML ما وجود دارد باید فولدری با نام images وجود داشته باشد و در آن فولدر عکسی با نام logo.gif موجود باشد.

```
src="../images/logo.gif"
```

یعنی تصویر ما یک سطح بالاتر از مسیر فایل HTML ما قرار دارد. یعنی اگر عکس ما در فولدری با نام images وجود دارد، در این فولدر، فولدر دیگری هم وجود دارد که فایل HTML ما در آن قرار دارد. توضیح دادن اینها کمی مشکل است خودتان به جوری متوجه شوید!! البته شما می‌توانید این مسیر را بوسیله یک URL مشخص کنید

```

```

- حالا استفاده از کدام بهتر است؟
  - اگر چه شاید استفاده از Url (Uniform Resource Locator) راحت‌تر باشد اما مزیت استفاده از آدرس‌دهی نسبی در این است که اولاً شما می‌توانید یک سایت را به صورت offline طراحی کنید و سپس همه آنرا همزمان روی سرور خود Upload کنید. در ثانی مرورگرها با آدرس‌های نسبی بهتر کار می‌کنند.
- **نکته مهم:** اگر سروری که با آن کار می‌کنید یک سرور یونیکس باشد، کوچک و بزرگ بودن حروف با هم تفاوت دارد مثلاً LOGO.GIF و logo.gif در ویندوز فرقی با هم ندارند ولی در یک سرور یونیکس اگر در خط دستورات LOGO.GIF را تایپ کنید ولی نام فایل شما logo.gif باشد عکسی به نمایش در نخواهد آمد. پیشنهاد این است که همیشه از حروف کوچک استفاده کنید و از فاصله هم در نام فایل‌ها استفاده نکنید. (البته همانطور پیشتر اشاره شد، در دستورات کوچک و بزرگی حروف تفاوتی نمی‌کند مگر در موارد خاص)
- توصیه می‌شود که حتماً ابعاد تصویر را با کمک شناسه‌های width و height قید کنید، اینکار سبب کمک به مرورگر در تسریع شروع نمایش صفحه خواهد شد. (در واقع با قید ابعاد تصویر، مرورگر قبل از لود تصویر می‌داند که چه مقدار از صفحه نمایش باید به تصویر اختصاص داده شود و قبل از لود شدن تمامی تصاویر مرورگر شروع به نمایش قالب و چارچوب کلی صفحه خواهد کرد.)
- می‌توانید یک تصویر را به صورت یک پیوند در آورید:

```
Go to
```

## نسخه ها (Versions) و DOCTYPE

در حال حاضر، HTML سه حالت دارد: Strict, Transitional, Frameset

حالت اول، یعنی حالت Strict، استفاده از المان‌های قدیمی را مجاز نمی‌داند. حالت دوم و سوم، هر دو استفاده از المان‌های قدیمی را پشتیبانی می‌کنند. تنها تفاوت آنها در این است که حالت سوم، به شما اجازه استفاده از فریم‌ها (Frames) را نیز می‌دهد.

برای تعیین حالت و نسخه‌ای که در سند خود بکار می‌برید، باید از اعلان DOCTYPE استفاده کنید. DOCTYPE اطلاعاتی را که برای برنامه‌های مرورگر، ضروری هستند ارائه می‌کنند.

### آغاز کردن یک صفحه HTML در حالت Transitional

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD/HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

### آغاز کردن یک صفحه HTML در حالت Frameset

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

### آغاز کردن یک صفحه HTML در حالت Strict

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD/HTML 4.01 Strict//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

## چگونگی Encode کردن صفحه

بهترین روش برای اینکه یک صفحه HTML را Encode کنید (این کار سبب می‌شوند تا کاراکترهای به کار رفته در صفحه وب شما، به درستی نمایش داده شوند)، استفاده از تگ meta استفاده نمایید. این تگ را باید در قسمت head تایپ نمایید:

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

اگر از استاندارد یونیکد برای فارسی نویسی استفاده میکنید، برای دیدن درست صفحات فارسی باید از نسخه ۵ Internet Explorer و به بالا و در مورد Netscape Navigator از نسخه ۶,۲ و به بالا استفاده کنید.

## المان‌های block-Lever و Inline

- یک المان می‌تواند به صورت Block-Lever یا Inline ظاهر شود.
- اگر به صورت Block-lever نوشته شود، همواره روی یک خط جدید قرار خواهد گرفت، درست مثل یک پاراگراف جدید در یک کتاب. اگر Inline باشد، بر روی خط فعلی نوشته می‌شود. مثل کلمه بعدی در یک پاراگراف.
- المان‌های Block-Lever، قطعات ساختاری بزرگ صفحه محسوب می‌شوند و به همین دلیل می‌توانند شامل المان‌های Inline، متن و المان‌های Block-Lever دیگر باشند. در مقابل المان‌های Inline فقط می‌توانند متن و المان‌های Inline دیگر را در بر بگیرند.