

آموزش کاربردی

HTML 5

و

CSS 3

آموزش کاربردی

**HTML 5**

و

**CSS 3**

مهندس محمد مرادی

انتشارات پندار پارس

سرشناسه	:مرادی، محمد، 1366 -
عنوان و نام پدیدآور	:آموزش کاربردی HTML5 و CSS3 /محمد مرادی.
مشخصات نشر	:تهران: پندار پارس، 1392.
مشخصات ظاهری	:xv، 312 ص: مصور (رنگی)، جدول.
شابک	:148000 ریال: 8-44-6529-600-978
وضعیت فهرست نویسی	:فیبا
یادداشت	:چاپ قبلی: پندار پارس: مانلی، 1390.
یادداشت	:چاپ دوم.
موضوع	:وب--سایتها-- طراحی
موضوع	:زبان نشانه‌گذاری فرامتنی
موضوع	:زبان نشانه‌گذاری فرامتنی توسعه‌پذیر
موضوع	:برگه‌های آشناری
رده بندی کنگره	:TK5105/51392888 418م/
رده بندی دیویی	:76/006
شماره کتابشناسی ملی	:3293785

### انتشارات پندار پارس



دفتر فروش: انقلاب، ابتدای کارگرجنوبی، کوچه رشتچی، شماره 14، واحد 16 [www.pendarepars.com](http://www.pendarepars.com)  
 تلفن: 66572335 - تلفکس: 66926578 همراه: 09122452348  
[info@pendarepars.com](mailto:info@pendarepars.com)

نام کتاب : آموزش کاربردی HTML5 و CSS3

ناشر : انتشارات پندار پارس

ترجمه و تالیف : محمد مرادی

چاپ نخست (ویرایش 2) : آبان 92

شمارگان : 1000 نسخه

طرح جلد : محمد اسماعیلی هدی

لیتوگرافی، چاپ، صحافی : ترام‌سنج، فرشویه، خیام

قیمت : 14800 تومان : شابک : 8-44-6529-600-978

\* هرگونه کپی برداری، تکثیر و چاپ کاغذی یا الکترونیکی از این کتاب بدون اجازه ناشر تخلف بوده و پیگرد قانونی دارد\*

## فهرست

1	.....	فصل 1 معرفی HTML و CSS
2	.....	چرا نیاز به یادگیری CSS و HTML داریم؟
3	.....	HTML یا XHTML؟
4	.....	شروع مثال
4	.....	چارچوب صفحه
9	.....	ایجاد Header
9	.....	ایجاد محتوا
12	.....	بخش پانوشت یا فوتر (footer)
23	.....	فصل 2 آشنایی اولیه با HTML 5 و CSS 3
23	.....	آشنایی با HTML5 و CSS3
26	.....	شروع مثال
30	.....	ایجاد بخش header
31	.....	ایجاد بخش ویژه (Featured)
32	.....	ایجاد بدنه اصلی طراحی
34	.....	ایجاد بلاک Extras
35	.....	بخش Footer
36	.....	شروع سبکدهی به وسیله CSS
38	.....	شرح چند ویژگی
39	.....	ایجاد سازگاری برای مرورگرهای قدیمی
39	.....	محدود کردن پهنای بلاکها
40	.....	سبکدهی Header
42	.....	سبکدهی بلاک ویژه (featured block) و بدنه اصلی
43	.....	سبکدهی بلاک Extras

44	افزودن آیکن‌های مربوط به شبکه‌های اجتماعی (Social Networks)
45	سبک‌دهی Footer
46	لیست پست‌ها (Posts)
47	حل مشکل IE
53	<b>فصل 3 ترسیم به‌وسیله عنصر &lt;CANVAS&gt;</b>
53	آشنایی با عنصر <canvas>
54	شروع مثال Canvas
55	رسم چهارضلعی‌ها
57	رسم خطوط هنری (Line Art)
59	پر کردن خطوط هنری
60	رسم توسط منحنی‌های بزییر (Bezier)
62	ترسیم منحنی‌های درجه دوم
63	رسم کمان‌ها
64	رسم متن
69	<b>فصل 4 ایجاد افکت کشیدن و رها کردن (DRAG &amp; DROP)</b>
69	مسائل اولیه
71	آشنایی با توابع مربوط به drag & drop
71	ویژگی draggable
72	ویژگی ondragenter
72	ویژگی ondragover
72	ویژگی ondrop
73	ویژگی ondragstart
73	ویژگی ondragend
73	شیء dataTransfer

74.....	شروع مثال drag and drop
76 .....	سبک‌دهی به عناصر draggable و مقصد (Target)
78 .....	شروع عملیات کشیدن
79 .....	اجازه دادن به عناصر کشیده شده برای ورود به مقصد (Target)
79 .....	اجازه دادن به اشیاء کشیده شده برای رها شدن در مقاصد مشخص
80 .....	مدیریت رویدادهای رها شدن (drop event)
81 .....	پایان دادن به عملیات رها شدن
85 .....	<b>فصل 5 کنترل‌های فرم</b>
86.....	آشنایی با فرم‌ها
89.....	آشنایی با توابع کنترل‌های فرم (webform API)
90.....	شروع کار بر روی مثال
91 .....	ایجاد کنترل Default
92 .....	ایجاد یک کنترل URL
93 .....	ایجاد یک کنترل Email
94 .....	ایجاد کنترل‌های عدد و محدوده (number&range)
95 .....	ایجاد کنترل‌های Date و Time
97 .....	ایجاد کنترل رنگ
98 .....	ایجاد کنترل (Search)
101 .....	<b>فصل 6 ویرایش درجا (INLINE EDITING)</b>
101.....	آشنایی با ویرایش درجا (inline editing)
101.....	contenteditable
102.....	designmode
102.....	spellcheck
102.....	شروع کار بر روی مثال editdiv.html

104.....	Bold	افزودن دکمه
106.....	Italic	افزودن دکمه
107.....	Underline	اضافه کردن دکمه
108.....	(Add Link)	افزودن دکمه اضافه کردن پیوند
111.....	Display Source	افزودن دکمه
113.....	(SpellChecking)	مطابقت املائی
117 .....	(VIDEO AND AUDIO)	فصل 7 استفاده از فایل‌های صوتی و تصویری
118.....	Video	شروع کار با کنترل
119.....	Video	آشنایی با توابع کار با عنصر
119.....	autoplay	ویژگی
119.....	controls	ویژگی
119.....	height	ویژگی
119.....	loop	ویژگی
119.....	poster	ویژگی
120.....	preload	ویژگی
120.....	src	ویژگی
120.....	width	ویژگی
121.....	onerror	ویژگی
121.....	ogg	تبدیل فایل‌های ویدئویی به قالب
122.....	video.html	شروع مثال
123.....	video.html	افزودن ویژگی controls به مثال
123.....		اجرای متوالی ویدئو
124.....		اجرای خودکار ویدئو

124.....	تشخیص عدم اجرای ویدئو
127.....	شروع کار با کنترل Audio
127.....	شروع مثال audio.html
131 .....	<b>فصل 8 ذخیره اطلاعات (WEBSTORAGE)</b>
131.....	Session Storage
133.....	ویژگی length
133.....	تابع key()
133.....	تابع getitem()
133.....	تابع setitem()
133.....	تابع removeitem()
134.....	تابع clear()
134.....	شروع مثال sessionstorage.html
134.....	ذخیره داده‌ها در session
135.....	بازیابی داده‌ها از session
136.....	پاک کردن اطلاعات از session
137.....	Local Storage
141 .....	<b>فصل 9 معرفی CSS3</b>
141.....	آشنایی با CSS3
142.....	border-radius
143.....	text-shadow
143.....	box-shadow
143.....	تصاویر چندگانه برای پس‌زمینه (Multiple Background Images)
143.....	opacity



144.....	RGBA
145.....	vendor-specific prefixes
146.....	پشتیبانی از مرورگرهای قدیمی
146.....	انتخاب کننده‌های CSS3
147.....	انتخاب کننده‌های رابطه‌ای
147.....	رابطه نوادگی (Descendent (E F))
148.....	Child(E>F)
148.....	هم‌نژادی مجاور یا Adjacent Sibling (E+F)
148.....	هم‌نژادی عمومی یا (General Sibling [E~F])
149.....	انتخاب کننده‌های ویژگی (Attribute Selectors)
149.....	E [attr]
149.....	E [attr=val]
149.....	E [attr  = val]
149.....	E [attr ~= val]
149.....	(IE8+, webkit, opera, mozilla), E [attr ^= val]
149.....	(IE8+, webkit, opera, mozilla), E [attr \$= val]
150.....	E [attr *= val]
150.....	کلاس‌های کاذب (pseudo-classes)
152.....	کلاس‌های کاذب ساختاری (Structural Pseudo-classes)
154.....	عناصر کاذب (pseudo elements)
154.....	رنگ‌ها در CSS3
155.....	RGBA
155.....	HSL, HSLA

156.....	میزان شفافیت (opacity)
157.....	گوشه‌های منحنی (Rounded Corners)
158.....	حاشیه‌های تصویری یا border images
159.....	Drop Shadow
160.....	Text Shadow
161 .....	<b>فصل 10 ایجاد افکت GRADIENT و پس‌زمینه چندگانه در CSS3</b>
161.....	چرا از واژه شبیه‌سازی استفاده کردیم؟
162.....	Gradient خطی
162.....	گرامر W3C برای ایجاد gradient
164.....	شیوه قدیمی Webkit
164.....	gradient خطی با استفاده از SVG
166.....	gradient خطی با استفاده از فیلترهای IE
166.....	ابزار کمکی
167.....	Radial Gradient
167.....	گرامر W3C
169.....	شیوه قدیمی Webkit
170.....	تکرار gradientها
171.....	تصاویر پس‌زمینه چندگانه
173.....	ویژگی background-size
175 .....	<b>فصل 11 ویژگی‌های TRANSITIONS و TRANSFORMS در CSS3</b>
175.....	Transforms
176.....	Translations
176.....	تغییر اندازه

177	چرخش (Rotation)
178	کج کردن (Skew)
178	تغییر مبدأ Transform
179	پشتیبانی از نسخه‌های قدیمی‌تر مرورگر IE
180	Transitions
181	Transition-property
182	Transition-duration
182	Transition-timing-function
183	Transition-delay
184	خلاصه‌نویسی ویژگی‌های Transition
184	انتقال‌های چندگانه (Multiple Transition)
185	Animations
186	keyframes
187	ویژگی‌های انیمیشن (Animation Properties)
187	animation-name
187	animation-duration
188	animation-timing-function
188	animation-iteration-count
188	animation-direction
188	animation-delay
189	animation-fill-mode
189	animation-play-state
189	خلاصه‌نویسی ویژگی‌های انیمیشن

191	فصل 12 کار با فونت‌ها و طراحی‌های چند ستونی
191	کار با فونت‌ها با استفاده از @font-face
192	پیاپیاده‌سازی @font-face
193	مشخص کردن منابع فونت
194	اعمال فونت‌ها
195	ایجاد فرمت‌های مختلف فونت
196	سایر ملاحظات
196	طراحی‌های چند ستونه (Multicolumn Layout)
197	ویژگی Column-Count
197	ویژگی Column-gap
198	ویژگی Column-width
198	ویژگی Column-Rule
201	فصل 13 تمرین پایانی
202	ایجاد ساختار اصلی
202	عنصر Doctype
203	عنصر <head>
203	ساختاردهی وب سایت
205	ایجاد فرم ثبت نام
207	سبکدهی به فیلدهای اجباری فرم
208	ویژگی Placeholder
208	افزودن الگو (Pattern)
209	افزودن ویدئو
210	ایجاد گوشه منحنی (Round Corners)

212.....	ایجاد Gradient
214.....	افزودن ویژگی‌های Transforms
214.....	افزودن فونت
215.....	ایجاد ستون‌های چندگانه
216.....	افزودن ویژگی Drag and Drop
219 .....	<b>پیوست الف عناصر جدید HTML 5</b>
220.....	عنصر <article>
222.....	عنصر <aside>
222.....	عنصر <command>
224.....	عنصر <datalist>
224.....	عنصر <details>
225.....	عنصر <embed>
226.....	عنصر <figcaption>
226.....	عنصر <figure>
227.....	عنصر <footer>
228.....	عنصر <header>
229.....	عنصر <hgroup>
229.....	عنصر <mark>
230.....	عنصر <meter>
231.....	عنصر <nav>
232.....	عنصر <progress>
233.....	عنصر <section>
233.....	عنصر <summary>

235	.....	<b>پیوست ب</b>	<b>مروری بر CSS</b>
235	.....	شروع کار با	..... CSS
239	.....	تشریح اجزای مختلف	..... CSS
240	.....	توضیحات (Comments)	.....
240	.....	روش‌های اعمال سبک‌دهی	.....
241	.....	واحدهای اندازه‌گیری	.....
241	.....	تعیین رنگ‌ها	.....
242	.....	روش‌های انتخاب عناصر مختلف برای سبک‌دهی	.....
243	.....	انتخاب کننده‌های کلاس‌های کاذب (Pseudo-class)	.....
244	.....	لیست‌ها	.....
246	.....	مشخص کردن یک خانواده فونت	.....
248	.....	تزئین متون	.....
249	.....	<b>پیوست ج</b>	<b>RESPONSIVE WEB DESIGN</b>
252	.....	اصول طراحی واکنش‌گرا	.....
257	.....	کارکرد Fluid Grids	.....
261	.....	طراحی سیال در قالب فریم‌ورک‌ها	.....
265	.....	تصاویر انعطاف‌پذیر (تطبیق‌پذیر)	.....
269	.....	Media Queries	.....
272	.....	محتوای واکنش‌گرا	.....
275	.....	ساختار سند	.....
276	.....	فراداده‌ها	.....
277	.....	محتوای پشتیبان	.....
277	.....	تمرینات	.....

## پیش‌گفتار

همواره یکی از بزرگترین دغدغه‌هایی که در رابطه با تکنولوژی‌های جدید وجود دارد، چگونگی ارائه آنها به کاربران و علاقه‌مندان است. در حقیقت این مسئله، چالشی در مقیاس جهانی است و می‌توان با مشاهده کتاب‌ها و مقالات نوشته شده در زمینه این‌گونه تکنولوژی‌ها به آن پی برد. چراکه هر یک از آنها تلاش می‌کنند تا به زعم خود به بهترین شکل ممکن، اطلاعات مورد نیاز را به خوانندگان ارائه نمایند. البته شیوه ارائه مطالب نیز تابع مخاطبی است که ناشران به عنوان مخاطب هدف در نظر می‌گیرند و این موضوع سبب می‌شود که برخی از این منابع تنها برای کاربرانی که در سطح پیشرفته قرار دارند مناسب بوده و برخی دیگر به علت سادگی، برای آنها خسته کننده باشد.

البته از یاد نبریم که بیشتر این آموزش‌ها نیز جنبه عملی و کاربردی نداشته و تنها به ذکر خصوصیات و ویژگی‌ها می‌پردازند.

در این کتاب ما به بحث در مورد دو تکنولوژی جدید و در واقع دو استاندارد آینده طراحی وب، یعنی نسخه‌های جدید و ارتقاء یافته زبان‌های HTML و CSS پرداخته‌ایم. با توجه به مسائل ذکر شده و نیز با تجاربی که در طول چندین سال بررسی و کار با تکنولوژی‌های این چینی توسط نگارنده کسب شده، در این کتاب تلاش شده است تا مطالب به شیوه‌ای متوازن ارائه گردد تا هر دو گروه مخاطب مبتدی و حرفه‌ای را پوشش دهد.

از آنجاکه به دلیل جدید و در حال بررسی بودن عناوین مورد بحث، نمی‌توان ادعا کرد که می‌توان برای آنها کتاب مرجعی در نظر گرفت، برآن بوده‌ایم در عین حال که به معرفی ویژگی‌های جدید افزوده شده می‌پردازیم، آنها را در حد امکان و به گونه‌ای که برای کاربر ایرانی قابل استفاده باشد بررسی نماییم. علاوه براین برای حفظ جنبه کاربردی این آموزش‌ها نیز در فصول آغازین و پایانی، تمرینات جامعی تدارک دیده شده تا کاربران از طریق آن بتوانند آموخته‌های خود را به صورت عملی به کار بندند. البته چون این امکان وجود نداشت که تمامی مباحث مربوط به نسخه‌های قدیمی HTML و CSS را در این کتاب بررسی کنیم، برای بهره بردن حداکثری از مطالب نیاز است که خوانندگان محترم، آشنایی هرچند جزئی با آنها داشته باشند.

روی هم رفته، با استعانت از خداوند متعال، در این کتاب تلاش شده است تا نکات و مسائل مرتبط با تکنولوژی‌های مورد بحث به صورت تکمیلی و کاربردی و در عین حال به دور از اطناب و زیاده‌گویی ارائه شود، به امید اینکه بتواند به خوانندگان در ایجاد صفحات وب مدرن و همگام با تکنولوژی‌های روز یاری رساند.

در انتها جا دارد از زحمات و مساعدت‌های جناب آقای مهندس یعسوبی، مدیریت محترم انتشارات پندار پارس که بدون همکاری ایشان این کتاب فراهم نمی‌شد، نهایت تقدیر و تشکر را داشته باشم. از آنجا که هیچ نوشته فنی نمی‌تواند خالی از اشکال و نقص باشد، شما خواننده گرامی از طریق آدرس [moradi.c85@gmail.com](mailto:moradi.c85@gmail.com) می‌توانید انتقادات، پیشنهادات و پرسش‌های خود را با ما در میان بگذارید.

پاییز 90

محمد مرادی



# فصل 1

## معرفی HTML و CSS

استفاده از HTML به عنوان یک زبان نشانه‌گذاری برای خلق صفحات وب، به نخستین روزهای پیدایش وب، یعنی اوایل دهه 90 میلادی باز می‌گردد. هم‌زمان با ابداع وب توسط متخصصین مرکز تحقیقات فیزیک اروپا<sup>1</sup>، دو ابزار HTML و مرورگر<sup>2</sup> نیز توسط آنها ابداع شد. HTML به‌عنوان زبانی برای ایجاد و ساختاردهی صفحات وب، و مرورگر به‌عنوان نرم‌افزاری به‌منظور تفسیر و نمایش اسناد نوشته شده به زبان HTML بود. در واقع HTML به ایجاد کنندگان صفحات وب این امکان را می‌داد که مطالب، عکس‌ها و پیوندهای<sup>3</sup> مورد نظر خود را با استفاده از قواعدی که برای زبان وجود داشت در صفحه قرار دهند و به عبارت دیگر، صفحات وب خود را طراحی یا ایجاد کنند. در آن سو، نوع به‌کارگیری این قواعد، به مرورگر اجازه می‌داد که نتیجه را به صورت صفحات وب در خروجی برای کاربران نمایش دهد. صفحات طراحی شده توسط نسخه‌های ابتدایی HTML از لحاظ بصری خیلی چشم نواز نبودند و همین امر سبب شد در طول سال‌ها، بنیاد جهانی وب (W3C) که متولی استانداردهای و ارائه نسخه‌های جدیدتر و نیز ایجاد تغییرات در HTML است، به منظور ارتقاء سطح کیفی صفحات وب، ویژگی‌های دیگری در HTML بگنجانند.

البته در اواخر دهه 90 میلادی استفاده از ویژگی‌های موجود در HTML به منظور آراستن صفحات وب به تدریج از دستور کار طراحان صفحات وب خارج شد. علت آن این بود که کارشناسان اعتقاد داشتند از آنجا که HTML مسئولیت نمایش اطلاعات در صفحات وب را به عهده دارد و این اطلاعات به خودی خود بسیار حائز اهمیت هستند، نباید از HTML به عنوان وسیله‌ای برای جلوه‌دهی به این اطلاعات استفاده کرد، بلکه باید از آن تنها به منظور اسکلت‌بندی و تعیین چارچوب یک صفحه وب استفاده کرد.

مجموعه این عوامل سبب شد که تکنولوژی دیگری در آن زمان به وجود آید و به علت قابلیت‌های فراوان و انعطاف پذیری بالای آن، به سرعت در بین طراحان وب محبوب شود. این تکنولوژی چیزی به جز CSS یا Cascading Style Sheet نبود.

---

<sup>1</sup> CERN

<sup>2</sup> Browser

<sup>3</sup> Links

CSS به طراحان اجازه می‌داد امور مربوط به ظاهر صفحات وب، شامل انتخاب و انتساب جایگاه عناصر، رنگ‌ها، سطح‌بندی عناصر در داخل صفحه و مواردی مانند این را به راحتی انجام دهند. علاوه بر این، چون این کار باعث می‌شود کدهای مربوط به ساختار اصلی صفحه از کدهای مربوط به شیوه ارائه جدا شود، سبب می‌شود هم حجم صفحات وب کاهش یابد (که این عمل باعث افزایش سرعت بارگذاری<sup>1</sup> صفحات می‌شد) و هم خوانایی کدهای به‌کار رفته را بیشتر می‌کند.

از آن پس CSS و HTML به‌عنوان دو یار جدا نشدنی در عرصه طراحی وب مورد استفاده قرار گرفتند.

در راستای ارتقاء توانمندی‌های این تکنولوژی‌ها، بنیاد جهانی وب نسخه‌های متعددی از آنها را مورد تصویب قرار داد که آخرین نسخه از آنها که به عنوان استاندارد کنونی وب مطرح بوده و توسط تمامی مرورگرهای استاندارد پشتیبانی می‌شود، HTML 4.01 و CSS 2.1 می‌باشد.

## چرا نیاز به یادگیری CSS و HTML داریم؟

همه طراحان وب و افرادی که قصد دارند یک وب سایت ایجاد کنند، چه متخصصان و حرفه‌ای‌ها در این زمینه و چه افرادی که از سیستم‌های مدیریت محتوا (CMS) و یا Blogها برای خلق صفحات وب استفاده می‌کنند، نیاز به دانشی هر چند جزئی از CSS و HTML دارند. چرا که حتی اگر از سیستم‌های مدیریت محتوا به هر صورت استفاده شود، مطمئناً مواردی پیش خواهد آمد که احتیاج به دستکاری کدها وجود خواهد داشت و این کار تنها از عهده افرادی برمی‌آید که آشنایی نسبی با این تکنولوژی‌ها دارند.

از این رو در ادامه این فصل به بررسی اجمالی چگونگی استفاده عملی از CSS و HTML می‌پردازیم. البته در این فصل قصد نداریم به آموزش HTML نسخه 4.01 و CSS نسخه 2.1 بپردازیم ولی به جهت یادآوری برای کاربرانی که تجربه استفاده از آنها را دارند و نیز به‌منظور مرور سریع برای کاربرانی که تاکنون از آنها استفاده نکرده‌اند، با استفاده از این تکنولوژی‌ها یک صفحه وب طراحی نموده و نکات ضروری مربوط به آنها را در خلال این تمرین بررسی و مرور می‌کنیم.

در طول سال‌هایی که نسخه‌های مختلف HTML و CSS از طرف W3C ارائه می‌شدند متأسفانه همه مرورگرها قادر به پشتیبانی کامل از تمام خصوصیات و ویژگی‌های آنها نبودند و معمولاً طراحان مجبور می‌شدند از ترفندهای مختلفی استفاده نمایند تا صفحات وب طراحی شده توسط آنها، در تمام مرورگرها به صورت یکسان به نمایش درآید. البته خوشبختانه در حال حاضر و با توجه به وجود

<sup>1</sup> Load

مرورگرهای استاندارد، تقریباً نسخه‌های جدید همه مرورگرها، تمامی ویژگی‌های HTML 4.01 و CSS2.1 را به صورت مناسبی پشتیبانی می‌کنند.

پس از طراحی هر صفحه وب (و یا وب سایت)، برای بررسی مطابقت آنها بر روی مرورگرهای مختلف و کسب اطمینان از اینکه کاربران با مرورگرهای مختلف، تصویر یکسانی از صفحه وب (وب سایت) طراحی شده را مشاهده می‌کنند، باید صفحات طراحی شده را در مرورگرهای مختلف تست کرده و نتایج را بررسی نماییم.



برای این کار دو شیوه وجود دارد:

روش نخست، تنزل مطبوع (graceful degradation) نام دارد. در این روش ابتدا طراحی برای عموم مرورگرها انجام می‌شود و در ادامه، از تکنیک‌هایی استفاده می‌شود تا بتوان نقص‌های موجود در یک مرورگر خاص را برطرف نمود. به عبارت دیگر در طراحی، ابتدا جدیدترین تکنولوژی‌ها مورد توجه قرار می‌گیرد و سپس به رفع نقص‌های مرورگرهای قدیمی‌تر پرداخته می‌شود.

روش دوم، ارتقاء تدریجی (progressive enhancement) نام دارد و اساس آن بر این اصل استوار است که ابتدا طراحی بر پایه مرورگری که ویژگی‌های خاص و قابلیت‌های کمتری دارد، مانند IE6 انجام گیرد، سپس ویژگی‌های دیگری به آن افزوده شود تا برای سایر مرورگرها قابل استفاده باشد.

انتخاب هر کدام از این شیوه‌ها بستگی به سلیقه و نظر طراح دارد و با توجه به نوع مسئله مورد نظر، هر یک از آنها می‌تواند گزینه مناسبی باشد.

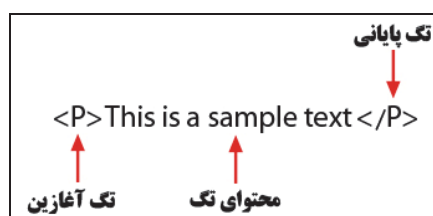
همان طور که پیش‌تر گفته شد اگرچه در این کتاب، قصد آموزش نسخه‌های پیشین HTML و CSS را نداریم ولی به جهت یادآوری و مرور، از طریق یک تمرین به بررسی آنها می‌پردازیم.

## HTML یا XHTML ؟

XHTML که مخفف Extensible HTML است، در واقع چیزی جز همان HTML نمی‌باشد و تنها تفاوت آن با HTML قواعد سفت و سخت‌تر آن در مورد استفاده از Tagهاست.

XHTML در دو نسخه متفاوت Strict و Transitional ارائه شده است. در نسخه نخست این امکان وجود ندارد که از Tagهایی که مربوط به جنبه‌های پیرایش و ظاهر صفحات وب هستند، استفاده کرد. مانند تگ <Font> ولی در نسخه دوم استفاده از آنها مجاز است.

به طور کلی HTML (X) از مجموعه‌ای از عناصر به نام Tag ایجاد شده و این تگ‌ها هستند که وظیفه ایجاد صفحات و مسائل پیرامون آن را دارند. تگ‌ها در این زبان، عناصر از پیش تعیین شده‌ای هستند که کاربردهای مخصوص به خود را دارند. متنی که در داخل تگ‌ها یعنی بین بخش آغازین و پایانی آن قرار می‌گیرد، تحت کنترل آن تگ قرار گرفته و هرگونه ویژگی که به آن تگ اعمال شود در واقع بر روی متن محصور در آن اعمال می‌شود.



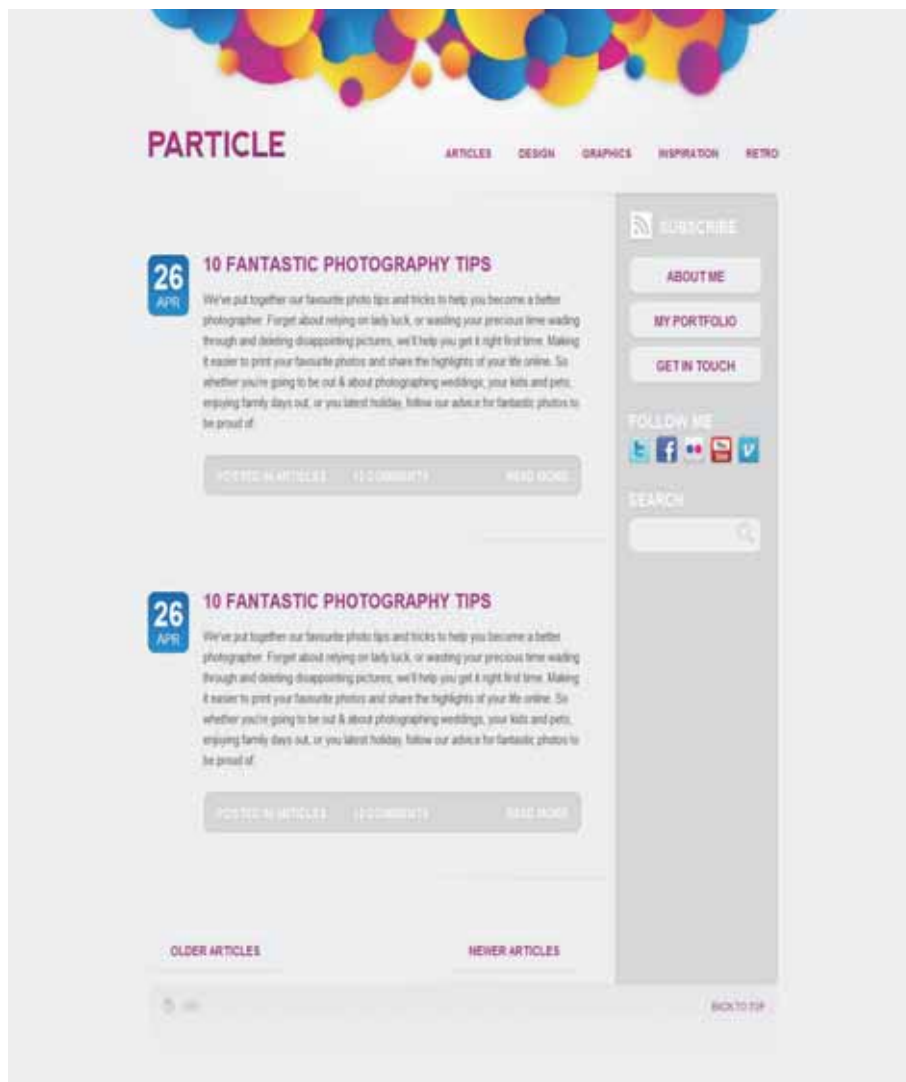
شکل 1-1 ساختار تگهای HTML

## شروع مثال

صفحه‌ای که ما برای نمونه قصد ایجاد آن را داریم، به صورت زیر می‌باشد. یعنی نتیجه کاری که در ادامه انجام می‌دهیم، قرار است صفحه وبی مشابه شکل 1-2 را ایجاد نماید.

## چارچوب صفحه

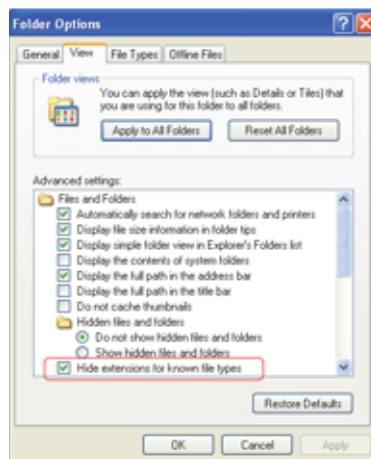
برای ایجاد هر صفحه (سند) HTML (X) ابتدا باید چارچوب اصلی آن را ایجاد کرد، سپس باقی عناصر و قسمت‌های مختلف را در آن قرار داد. برای ایجاد و کدنویسی HTML، نرم‌افزارها و ویرایش‌گرهای مختلفی وجود دارد که برخی از آنها مانند Adobe Dreamweaver امکانات زیادی را در اختیار کاربران قرار می‌دهد ولی بیشتر طراحان حرفه‌ای، از ویرایش‌گرهای بسیار ساده مانند نرم‌افزار Note Pad موجود در ویندوز استفاده می‌کنند و گدهای مربوط به پروژه‌های خود را به صورت دستی و به طور کامل کدنویسی می‌کنند.



شکل 2-1 تصویر نتیجه پایانی تمرین این فصل

برای ایجاد یک سند HTML (x) از طریق Note Pad باید مانند شکل زیر عمل کرده و گزینه مورد نظر را غیر فعال کرده (تیک آن را برمی‌داریم) سپس یک فایل متنی با نام دلخواه و فرمت HTML ایجاد کنید.





شکل 3 - 1 تنظیمات لازم برای ایجاد فایل‌ها با فرمت HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>Particle</title>

<link href="style.css" rel="stylesheet" type="text/css" media="screen" />

</head>

<body>
  <div id="container">

  </div>


</body>
</html>
```

در قطعه کد بالا، دو خط اول مربوط به معرفی نسخه استفاده شده HTML (X) برای ایجاد صفحه موردنظر می‌باشد. اگرچه بدون نوشتن این دو خط هم می‌توان یک صفحه HTML (X) ایجاد نمود ولی استفاده از این کدها دو ویژگی مهم دارد:

نخست اینکه، این خطوط به مرورگرهای مختلفی که صفحه قرار است بر روی آنها به نمایش درآید اعلام می‌کند که در کدنویسی این صفحه، از قواعد مربوط به کدام نسخه HTML (X) استفاده شده تا مرورگرها بر طبق آن اقدام به تفسیر کدها و نمایش خروجی کنند. در چنین مواقعی خطر امکان اجرا نشدن برخی قسمت‌های کد بر روی مرورگرهای مختلف و مسائلی از این دست، بسیار کاهش می‌یابد.

ویژگی دوم استفاده از این کدها در واقع گسترش یافته‌ی ویژگی قبلی است. یعنی زمانی که قصد دارید سند HTML (X) خود را توسط نرم‌افزاری که در وب سایت W3C.org موجود است اعتبارسنجی کنید، اعتبار و صحت صفحه شما بر اساس نسخه‌ای از HTML که مورد استفاده قرار گرفته و در ابتدای سند به آن اشاره شده سنجیده می‌شود.

اعتبارسنجی اسناد HTML (X) و CSS که در وب سایت W3C.org و در آدرس‌های زیر انجام می‌شود، به صفحات ایجاد شده کمک می‌کند و به نوعی تضمین می‌کند که آنها در تمامی مرورگرهای استاندارد به خوبی و بدون هیچ مشکلی اجرا می‌شوند. همچنین این اعتبارسنجی، اشکالات موجود در کدهای نوشته شده را مشخص می‌کند و امر اصلاح آنها را تسهیل می‌نماید.



برای اعتبار سنجی اسناد HTML می‌توانید به آدرس زیر مراجعه نمایید:

<http://validator.w3.org>



شکل 4 - 1 نرم افزار اعتبار سنج HTML

همچنین برای اعتبارسنجی اسناد CSS می‌توان از نرم‌افزار اعتبارسنج (Validator) موجود در آدرس زیر استفاده کرد:

<http://jigsaw.w3.org/CSS-validator>



شکل 4 - 1 نرم افزار اعتبار سنج CSS

یک سند HTML (X) از سه بخش اصلی تشکیل شده است:

تگ `<HTML>` /`</HTML>` که اصلی‌ترین تگ می‌باشد و تمامی تگ‌های دیگر باید در بین این دو تگ قرار گیرد. قسمت دوم، تگ `<Head>` /`</Head>` می‌باشد. در این تگ چندین بخش مهم قرار می‌گیرد از جمله متاگ‌ها<sup>1</sup> که به وسیله آنها می‌توان اطلاعات اضافی در مورد صفحه وب موجود را به سند اضافه کرد. به عنوان مثال، متاگ‌های `Robot` و `Keyword` و مانند آن.

عموماً محتوای متاگ‌ها برای موتورهای جستجو قابل استفاده هستند و در خروجی مرورگرها به نمایش در نمی‌آیند. استفاده از آنها نیز اجباری نیست، ولی غالباً به دلیل فواید زیادشان مورد استفاده قرار می‌گیرند.



مورد بعدی، تگ `<title>` می‌باشد که متنی را که در بین آن نوشته می‌شود به صورت عنوان صفحه وب در گوشه سمت چپ و بالای پنجره مرورگر نمایش می‌دهد. کاربرد دیگر و مهم تگ `Head` این است که فایل‌های دیگری که نیاز هستند به یک صفحه HTML (X) الصاق شوند باید در این قسمت ذکر شوند. عموماً این فایل‌ها، فایل‌های CSS یا جاوا اسکریپت می‌باشند.

سومین بخش مهم یک سند HTML (X)، تگ `<Body>` /`</Body>` می‌باشد. تمامی محتوا و مطالبی که در یک صفحه وب به نمایش در می‌آید، شامل تصاویر، پیوندها و متون، همگی باید در این بخش از سند قرار گیرند.

در این بخش از مثال در قسمت `body` صفحه تنها از یک عنصر `<div>` استفاده کرده‌ایم و در ادامه، سایر قسمت‌های لازم را به آن اضافه می‌نماییم.

عنصر `div` یکی از عناصر پرکاربرد HTML (X) می‌باشد که به طراحان اجازه می‌دهد با استفاده از آن، صفحه خود را به قسمت‌های مختلف تقسیم نمایند و محتوای مورد نظر خود را در آن قرار داده و از این طریق کنترل بیشتری بر روی بخش‌های مختلف صفحه داشته باشند.

همان‌طور که در عنصر `div` مشاهده می‌شود، ویژگی به نام `id` در آن به کار برده شده است. این ویژگی به هر عنصر، یک نام منحصر به فرد اختصاص می‌دهد که از طریق آن نام، عنصر مورد نظر توسط کدهای CSS قابل دستیابی می‌شود و تغییراتی که نیاز است بر روی آن ایجاد شود، شامل تغییر در مورد جایگاه قرارگیری در صفحه، اندازه فونت و امثال آن بر آن اعمال می‌گردد.

<sup>1</sup> Meta Tag



## ایجاد Header

```
<div id="header">
  <h1><a href="#"></a></h1>

  <ul id="categories">
    <li><a href="#">Articles</a></li>
    <li><a href="#">Design</a></li>
    <li><a href="#">Graphics</a></li>
    <li><a href="#">Inspiration</a></li>
    <li><a href="#">Retro</a></li>
  </ul>
</div>
```

در این بخش ابتدا به ایجاد header یا قسمت بالای صفحه که شامل لوگو و آیتم‌های مربوط به منوی کاربری می‌باشد می‌پردازیم. لازم به تذکر است که تمامی کدهای این قسمت در داخل تگ div بخش قبل با ID=container قرار می‌گیرند.

به این منظور از یک عنصر div دیگر با header = id استفاده می‌کنیم. اسامی که برای IDها انتخاب می‌کنیم از هیچ قاعده مشخص و ثابتی پیروی نمی‌کنند و انتخاب آنها کاملاً به سلیقه طراح بستگی دارد، ولی بیشتر اوقات برای سادگی کار و نیز راحتی یادآوری، از نام‌هایی استفاده می‌شود که متناسب با کاربردها باشند. پس از قرار دادن تصویر مربوط به لوگوی صفحه در ابتدا، برای ایجاد آیتم‌های منو از عنصر <UL> که مخفف unordered list یا لیست نامرتب است استفاده می‌شود. تقریباً استفاده از این روش برای ایجاد منوهای کاربری یک روش استاندارد در بین همه طراحان در سراسر جهان به شمار می‌رود.

با استفاده از عنصر <UL> /<UL> که در داخل خود می‌تواند هر تعدادی از عناصر <li> /<li> را قرار دهد، این امکان به طراحان داده می‌شود که تنها با اعمال ویژگی به عنصر <UL> بر روی تمامی عناصر زیر مجموعه آن، آن اثر یا ویژگی را اعمال نمایند.

برای هر یک از آیتم‌های منو که به صورت یک پیوند (link) هستند، باید مقصدی را مشخص کرد که آن مقصد عموماً آدرس صفحه‌ای دیگر می‌باشد ولی در اینجا چون ما صفحه دیگری برای این منظور نداریم، ویژگی href را در تگ <a> /<a> مساوی با کاراکتر # قرار می‌دهیم و این به این معناست که با کلیک بر روی آن آیتم، هیچ اتفاقی رخ نمی‌دهد.

## ایجاد محتوا

```
<div id="content">
  <div id="main"></div>
  <div id="side"></div>
</div>
```

در ادامه مثال خود، برای ایجاد محتوای صفحه از یک تگ div دیگر با ID = content استفاده می‌کنیم و در داخل آن از دو تگ div دیگر برای ایجاد محتوای بخش اصلی و دیگری برای ایجاد محتوای بخش کناری صفحه (side) استفاده می‌کنیم. در ادامه به ایجاد سایر قسمت‌های مورد نظر در این بخش می‌پردازیم.

```
<div id="main">
  <div class="post">
    <div class="date">
      <p>26 <span>Apr</span></p>
    </div>

    <div class="post-content">
      <h2><a href="#">10 Fantastic Photography Tips</a></h2>
      <p>We've put together our favourite photo tips and tricks to help you become a better photographer. Forget about relying on lady luck, or wasting your precious time wading through and deleting disappointing pictures, we'll help you get it right first time. Making it easier to print your favourite photos and share the highlights of your life online. So whether you're going to be out & about photographing weddings, your kids and pets, enjoying family days out, or you latest holiday, follow our advice for fantastic photos to be proud of.</p>

      <ul class="post-meta">
        <li>Posted in <a href="#">Articles</a></li>
        <li><a href="#">12 Comments</a></li>
        <li class="read-more"><a href="#">Read more</a></li>
      </ul>
    </div>

    <div class="pagination">
      <p class="older"><a href="#">Older articles</a></p>
      <p class="newer"><a href="#">Newer articles</a></p>
    </div>
  </div>
```

در این بخش قسمت‌های مختلفی که برای یک بلاگ مورد نیاز است را با استفاده از تگ‌های تودرتو ایجاد می‌کنیم، که نحوه استفاده از آنها به صورت واضحی مشخص است. تنها نکته مهم در این بخش استفاده از ویژگی Class برای برخی از عناصر می‌باشد. در واقع این ویژگی بسیار شبیه ویژگی ID است و همان عملکرد یعنی انتساب یک نام ویژه به یک عنصر را دارد ولی برخلاف آن، ویژگی Class باعث می‌شود که ویژگی‌های اعمال شده به یک عنصر با یک Class مشخص، به هر تعداد از عناصری که مقدار این ویژگی آنها مشابه است نیز اعمال گردد. به عبارت دیگر برای نام‌گذاری یک عنصر خاص، از ویژگی ID، و برای نام‌گذاری مجموعه‌ای از عناصر، ویژگی Class

به کار می‌رود. پس از انجام کارهای مربوط به بخش اصلی صفحه وب، قسمت‌های لازم برای بخش کناری (Side) را نیز به صورت زیر ایجاد می‌کنیم.

```
<div id="side">
  <p id="subscribe"><a href="#">Subscribe</a></p>

  <ul id="pages">
    <li><a href="#">About me</a></li>
    <li><a href="#">My portfolio</a></li>
    <li><a href="#">Get in touch</a></li>
  </ul>

  <h3>Follow me</h3>
  <ul id="follow-me">
    <li><a href="#" class="twitter">Twitter</a></li>
    <li><a href="#" class="facebook">Facebook</a></li>
    <li><a href="#" class="flickr">Flickr</a></li>
    <li><a href="#" class="youtube">YouTube</a></li>
    <li><a href="#" class="vimeo">Vimeo</a></li>
  </ul>

  <h3>Search</h3>
  <form id="search" method="get" action="">
    <fieldset>
      <input type="text" class="search-bar" />
      <input type="submit" value="Search" class="search-btn" />
    </fieldset>
  </form>
</div>
```

از عنصر <ul> زمانی استفاده می‌شود که قصد داشته باشیم تعدادی آیتم‌های مشابه و در کنار یکدیگر ایجاد کنیم. سپس از طریق قواعد CSS می‌توانیم نوع قرارگیری آنها (عمودی یا افقی) و سایر ویژگی‌های مورد نظر را بر روی آنها اعمال کنیم.

با استفاده از عناصر H1 تا H6 می‌توانیم عناوین را در صفحه ایجاد نماییم. هر چه مقدار این عدد کوچکتر باشد، سایز آن عنوان بزرگتر است.

یکی از عناصر پر استفاده در ایجاد صفحات وب، فرم‌ها هستند.

تقریباً همه روزه، حین استفاده از وب سایت‌های مختلف با آنها روبرو می‌شویم. این فرم‌ها می‌توانند مانند فرم جستجو در موتور جستجوی گوگل بسیار ساده باشند و یا همانند فرم‌های ثبت نام در یک انجمن اینترنتی، بسیار پیچیده باشند.

در این مثال، برای ایجاد کادر جستجو، از فرم‌ها استفاده می‌کنیم. فرم‌ها چند ویژگی مهم دارند که لازم است پیرامون آنها صحبت شود.

ویژگی نخست آنها method است که تعیین می‌کند اطلاعات چگونه به سمت سرور می‌دهند (Server) ارسال شود. این ویژگی می‌تواند دو مقدار post و get را بپذیرد که به علت مسائل امنیتی بهتر است از مقدار post برای آن استفاده شود.

ویژگی دیگر آن Action است که معمولاً آدرس صفحه‌ای در سمت سرور می‌باشد که وظیفه اعتبارسنجی و ثبت اطلاعات فرم را به عهده دارد. به عنوان مثال، مقدار این ویژگی می‌تواند به صورت زیر باشد:

```
Action="Submit.php "
```

برای ایجاد تکست‌باکس‌ها و دکمه‌های مختلف در داخل یک عنصر فرم، باید همگی آنها در داخل عنصر `<fieldset>` قرار گیرند. در این مثال، ما تنها به ایجاد یک تکست‌باکس و یک دکمه می‌پردازیم.

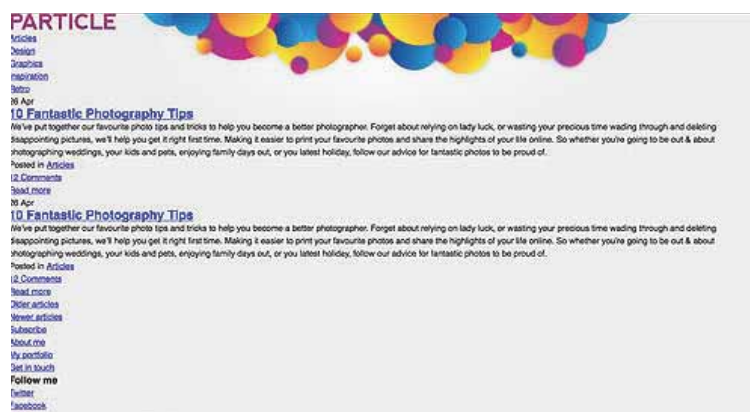
برای یک فرم، کنترل‌های مختلفی نیز می‌توان ایجاد کرده و نوع آنها را از طریق ویژگی type مشخص نمود: Radio برای دکمه‌های رادیویی، reset برای دکمه‌ای که اطلاعات موجود در فرم را پاک می‌کند و text area برای ایجاد کادرهای بزرگ متن.

## بخش پانویشت یا فوتر (footer)

```
<div id="footer">
  <ul id="credits">
    <li><a href="http://wordpress.org" class="wordpress">Powered by WordPress</a></li>
    <li><a href="http://www.blog.spoongraphics.co.uk" class="spoongraphics">Theme by SpoonGraphics</a></li>
  </ul>
  <p id="back-top"><a href="#header">Back to top</a></p>
</div>
```

در این بخش نیز همانند قسمت‌های قبلی به ایجاد موارد لازم برای قسمت پایانی صفحه می‌پردازیم. نکته قابل ذکر در این قسمت، خط پایانی کد می‌باشد. از این روش برای مواردی استفاده می‌شود که طول صفحات زیاد است. با ایجاد یک پیوند در پایین صفحه که اشاره به بالای صفحه دارد، کاربر می‌تواند با کلیک بر روی آن به سادگی، به بخشی از صفحه که ID آن در بخش href قرار دارد (در اینجا header) برود.

شکل زیر نمایش دهنده نتایج کارهایی است که تا این بخش انجام داده‌ایم.



شکل 6 - 1 نمایش نتیجه کدهای HTML نوشته شده برای ایجاد صفحه مورد نظر

همان طور که مشاهده می‌شود صفحه ما چندان زیبا و مناسب به نظر نمی‌آید. در این مرحله است که CSS وارد عمل می‌شود و وظیفه آراستن صفحه را به عهده می‌گیرد.

```
body, div, h1, h2, h3, h4, h5, h6, p, ul, ol, li, dl, dt, dd, img, form, fieldset, input, blockquote {
    margin: 0; padding: 0; border: 0;
}
```

```
body {
    font-family: Helvetica, Arial, Sans-Serif; line-height: 24px;
    background: #eee url(images/body-bg.jpg) center top no-repeat;
}
```

```
#container {
    width: 925px; margin: 0 auto; padding: 143px 0 0 0;
}
```

دو خط اول کد به این منظور است که سبک‌دهی (styling) را که برخی مرورگرها به صورت پیش فرض رعایت می‌کنند خنثی ساخته و از بروز مشکلات احتمالی در نمایش صفحه جلوگیری کند.

در مورد نحوه سبک‌دهی با استفاده از CSS، دو شیوه وجود دارد. شیوه نخست استفاده از نام‌های استاندارد تگ‌ها است، مثلاً در این قطعه کد می‌توانید مشاهده کنید که از نام `body` در خط اول استفاده شده است. در این شرایط، آن تگ مشخص با استفاده از ویژگی‌هایی که در میان دو گروهه مقابل آن ذکر شده، سبک‌دهی می‌شود.

روش دیگر، سبک‌دهی به عناصر با استفاده از نام‌هایی است که از طریق ویژگی‌های `ID` و `class` به آنها اطلاق می‌شود. برای اعلان استفاده از نام `ID`، باید در پشت نام مورد نظر، از علامت `#` استفاده نموده و زمانی که قصد سبک‌دهی به `class` را داریم از علامت `.` (نقطه) در پشت نام آن استفاده کنیم.

اگر در مواقعی از طریق ویژگی ID نیز نامی را به عنصری (مثل body) منتسب نموده و با استفاده از هر دو روش گفته شده ویژگی‌هایی را به آنها اعمال کنیم، در صورت تکراری بودن ویژگی‌های اعمال شده، آن ویژگی که توسط نام ID اعمال شده، اولویت بیشتری داشته و به عنصر اعمال می‌گردد.

```
#header {
    background: url(images/header-bg.png) right bottom no-repeat; overflow: hidden;
    padding: 0 0 50px 0;
}

#header h1 {
    float: left;
}

#header ul#categories {
    float: right; list-style: none; margin: 16px 0 0 0;
}

#header ul#categories li {
    float: left; margin: 0 0 0 40px;
}

#header ul#categories li a {
    display: block;
    font-size: 14px; font-weight: bold; text-transform: uppercase;
    color: #a42988;
    text-shadow: 0 2px 0px #fff; text-decoration: none;
}

#header ul#categories li a: hover {
    color: #006ab1;
}
```



شکل 7 - 1 نتیجه اعمال اولین بخش از قواعد CSS به طراحی

در ادامه با افزودن هر ویژگی به CSS، خروجی به وجود آمده در آن مرحله را نیز نمایش می‌دهیم تا متوجه اعمال تغییرات در آن شویم.