

HTML5

در قالب پروژه

CSS3

پیشن نیاز : HTML 4 و CSS 2



مؤلفان : الکسیس گلدستین ، لوییس لازاریس ، استلاویل
مترجم : امیر عباس عبدالعلی



آموزش HTML5 و CSS3 در قالب پروژه

مؤلفان: الکسیس گلدستین، لوییس لازاریس، استلا ویل

مترجم: امیرعباس عبدالعلی

پیش‌نیاز: HTML4 و CSS2.1



اینبار برای معرفی کتاب سعی کردیم از دوستان طراح، برنامه نویس، وبلاگ نویس و وب مستر برای نشر و معرفی این کتاب کمک بخواهیم. بدون کمک این دوستان شانس زیادی برای معرفی اثرمان وجود نداشت. در ادامه لوگوی و آدرس سایت‌هایی به ما لطف داشتند و در معرفی این اثر به ما کمک کرده‌اند، آورده شده است. از تمامی دوستانی که ما را در معرفی و نشر این فایل کمک کردند قدردانی می‌کنیم.



جامعه مجازی یا فاجعه مجازی - جواد احمدزاده

هر گونه خبر جدید درباره ی این کتاب (ارائه محتوای تکمیلی، فیلم و یا دوره آموزشی مرتبط با موضوع) در صفحه مربوط به کتاب منتشر می شود. برای کسب اطلاعات بیشتر درباره ی این کتاب می توانید به صفحه اصلی آن در سایت WeDesign مراجعه کنید

<http://books.wedesign.ir/html5-css3.html>

شبکه های اجتماعی

با دنبال کردن مترجم در شبکه های اجتماعی، می توانید مطالب بیشتری درباره ی طراحی وب سایت و تکنولوژی های مرتبط با آن به دست بیاورید.



خبرنامه ی WeDesign

علاوه بر آن می توانید با عضویت در خبرنامه ی سایت ما که در صفحه اول سایتمان در آدرس www.wedesign.ir قرار دارد، جدیدترین خبرها درباره ی کتاب های در دست تالیف و ترجمه، دوره های آموزشی و یا کارگاه هایی که قرار است برگزار کنیم را دریافت کنید. ایمیل شما نزد ما محفوظ است و به هیچ عنوان برای شما spam ارسال نمی کنیم.

فهرست مطالب

35	طراحی برای شرایط واقعی	11	درباره‌ی نویسندگان
فصل دوم: زبان نشانه‌گذاری و سبک‌های HTML5		13	مقدمه
37		14	معرفی گروه WeDesign
37	معرفی سایت HTML5 herald	14	محتوایات DVD کتاب
38	یک قالب اولیه برای HTML5	15	پیشگفتار
39	نوع سند	17	مقدمه کتاب
40	برچسب HTML	17	این کتاب برای چه کسانی مناسب است؟
41	برچسب head	18	این کتاب حاوی چه مطالبی است؟
42	هموار کردن زمین بازی	21	عرف‌های رعایت شده در این کتاب
45	بقیه‌ی برچسب‌ها	22	نمونه کدها
45	سؤال‌های رایج درباره‌ی HTML5	23	نکته‌ها، ترفندها و هشدارها
چرا این تغییرات همچنان در مرورگرهای		25	فصل اول: معرفی HTML5 و CSS3
46	قدیمی نیز کار می‌کند؟	25	HTML5 چیست؟
آیا همه برچسب‌ها در HTML5 باید بسته		27	چطور به اینجا رسیدیم؟
48	شوند؟	می‌شود مشخص کنید که کدام نسخه از	
دیگر قوانین کدنویسی XHTML چه می‌شود؟		27	مستندات HTML5 واقعی است؟
48		29	چرا HTML5 باید برایم مهم باشد؟
50	تعریف ساختار صفحه	30	CSS3 چیست؟
51	عنصر header	30	چرا CSS3 باید برایم مهم باشد؟
51	عنصر section	32	منظور ما از یادگیری در قالب پروژه چیست؟
53	عنصر article	32	بازار مرورگرهای مختلف
54	عنصر nav	34	رشد سریع وب‌گردی از طریق تلفن همراه
56	عنصر aside		
57	عنصر footer		
58	تعیین ساختار برای سایت HTML5 herald		
61	جمع‌بندی		

98	خاصیت <code>readonly</code>
98	خاصیت <code>multiple</code>
99	خاصیت <code>form</code>
99	خاصیت <code>autocomplete</code>
100	برچسب <code>datalist</code> و خاصیت <code>list</code>
101	خاصیت <code>autofocus</code>
نوع داده‌های ورودی جدید در فرم‌های HTML5	
101	
102	<code>search</code>
103	آدرس‌های <code>Email</code>
105	<code>URL</code> ها
105	شماره تلفن‌ها
106	اعداد
107	محدوده‌ها یا همان <code>ranges</code>
108	رنگ‌ها
109	تاریخ و ساعت
فیلدهای ورود داده‌ی دیگر که در HTML5 معرفی شده‌اند	
112	عنصر <code>output</code>
113	عنصر <code>keygen</code>
تغییرات در عناصر و خاصیت‌های مربوط به فرم که از نسخه‌های قبلی HTML وجود داشتند.	
113	
113	عنصر <code>form</code>
114	عنصر <code>optgroup</code>
114	عنصر <code>textarea</code>
114	نتیجه‌گیری

115 فصل پنجم: صوت و تصویر در HTML5

115	اندکی کنکاش در تاریخ
116	وضعیت در حال حاضر
117	فرمت‌های ویدئویی دربرگیرنده
117	شیوه‌های رمزگذاری / رمزگشایی فیلم
117	شیوه‌های رمزگذاری / رمزگشایی صوت
چه ترکیبی از فرمت‌های مختلف، در مرورگرهای موجود کار می‌کنند؟	
117	
118	کدنویسی
119	فعال کردن کنترل‌های بومی

63 فصل سوم: عناصر معناگرای دیگر

63	دید جدید نسب به نوع محتوا
65	رئوس مطالب در سند
68	خبر فوری
68	عنصر <code>hgroup</code>
70	عناصر جدید دیگر
70	عناصر <code>figure</code> و <code>figcaption</code>
71	عنصر <code>mark</code>
72	عناصرهای <code>meter</code> و <code>progress</code>
73	عنصر <code>time</code>
75	تغییرات در ویژگی‌های موجود
75	کلمه‌ی «نامناسب» دیگر نباید استفاده شود
76	عناصر بلوک در داخل برچسب پیوند
76	متن با فونت ضخیم
77	متن مایل
78	متن ریز و درشت
78	برچسب <code>cite</code> برای چشمان مجروح
79	لیست‌های توصیفی (نه لیست‌های تعریفی)
79	عناصر و قابلیت‌های جدید دیگر
79	عنصر <code>details</code>
80	لیست شماره‌دار سفارشی شده
80	سبک‌های متمرکز
81	خاصیت <code>async</code> برای برچسب <code>script</code>
82	تأیید اعتبار اسناد HTML5
84	خلاصه

85 فصل چهارم: فرم‌ها در HTML5

86	ابزارهای قابل استفاده در جعبه‌ابزار ما
88	خاصیت‌های مربوط به فرم در HTML5
88	خاصیت <code>required</code>
91	سبک‌دهی به فیلدهای ضروری فرم
92	خاصیت <code>placeholder</code>
Polyfillهای جایگزین خاصیت‌های <code>placeholder</code> که با جاوااسکریپت ایجاد شده-اند	
93	اند
96	خاصیت <code>pattern</code>
97	خاصیت <code>disabled</code>

158	عناصر کاذب و محتوای تولید شده	120	خصیصه‌ی autoplay
158	محتوای تولیدشده	121	خصیصه‌ی loop
159	::selection	121	خصیصه‌ی preload
160	رنگ‌ها در CSS3	122	خصیصه‌ی poster
160	RGBA	122	خصیصه‌ی audio
161	HSL و HSLA		پشتیبانی از چندین فرمت تصویری مختلف
162	ماتی	122	
163	استفاده از دستورات در عمل	124	ترتیب برچسب‌های source
165	گوشه‌های گرد با border-radius		تکلیف اینترنت اکسپلورر 6 تا 8 چه می‌شود؟
169	ایجاد سایه برای عناصر	125	
	استفاده از کلمه‌ی کلیدی inset و داشتن چند	126	MIME type ها
172	سایه به طور همزمان		تبدیل فیلم‌ها به فرمت قابل پخش بر روی وب
173	سایه برای متن با text-shadow	128	
174	سایه‌های بیشتر	128	ساخت کنترل‌کننده‌های اختصاصی
175	در فصل بعد می‌خوانیم		مقداری کد HTML و CSS برای شروع کار
فصل هفتم: توالی رنگ با CSS3 و تصاویر		129	
177	پس‌زمینه‌ی متعدد		معرفی API مربوط به عناصر چندرسانه‌ای
178	توالی رنگ خطی	131	
180	دستور استاندارد W3C	133	پخش و توقف فیلم
	دستور قدیمی در مرورگرهای برپایه‌ی	136	پخش و قطع صدای فیلم
184	WebKit		انجام عملی پس از به پایان رسیدن فیلم
185	کنار هم قرار دادن همه کدها	138	به‌روزرسانی زمان فیلم در زمان پخش آن
	ساخت توالی رنگ خطی با استفاده از SVG	138	
188			قابلیت‌های دیگر مربوط به API عناصر
	توالی رنگ خطی با استفاده از فیلترهای	141	چندرسانه‌ای
190	اینترنت اکسپلورر	141	رویدادها
190	ابزارهای مفید	142	خصیصه‌ها
191	توالی رنگ حلقوی	143	تکلیف صوت چیست؟
192	الگوی نگارش دستور به سبک W3C	144	محتوای صوتی و تصویری در دسترس
194	شیوه‌ی نگارش دستور قدیمی WebKit	144	اکنون زمان نمایش است
	هر چه یاد گرفته‌ایم را در کنار هم قرار دهیم		
196		147	فصل ششم: معرفی CSS3
197	توالی رنگ تکرارشونده	147	پشتیبانی از مرورگرهای قدیمی
199	استفاده از چندین تصویر در پس‌زمینه	148	گزینه‌ی‌گرهای CSS3
202	سایز تصویر پس‌زمینه	149	گزینه‌ی‌گرهای خویشاوندی
204	سخن پایانی	151	گزینه‌ی‌گرهای خصیصه
		152	کلاس‌های کاذب
		155	کلاس‌های کاذب ساختاری

230	تعیین مکان قرارگیری فایل فونت
233	توصیف‌گرهای خصوصیت فونت
234	محدوده Unicode
235	استفاده از فونت پیوست شده به صفحه
235	ملاحظات قانونی
	ساخت فونت با فرمت‌های مختلف از طریق
237	سایت Font Squirrel
240	ملاحظات دیگر
242	چیدمان چندستونه در CSS3
243	خصوصیت column-count
243	خصوصیت column-gap
244	خصوصیت column-width
	خصوصیت columns برای خلاصه‌نویسی
246	خصوصیات مربوط به ستون‌ها
247	ستون‌ها و خصوصیت height
248	خصوصیات دیگر مربوط به ستون‌ها
248	خصوصیت column-rule
248	ایجاد شکستگی در ستون
249	Spanning columns
250	ملاحظات دیگر
251	بهبود تدریجی
252	Media Queryها
253	Media queryها چه هستند؟
253	شیوه نگارش
254	Media queryها و انعطاف‌پذیری
255	پشتیبانی مرورگرها
255	مطالعه‌ی بیشتر
256	با طراحی زندگی کنید
فصل دهم: موقعیت مکانی، برنامه‌های تحت وب	
خارج از شبکه و ذخیره‌سازی تحت وب 259	
260	موقعیت مکانی
	نگرانی‌های مربوط به حفظ حریم شخصی
261	
262	متدهای سرویس موقعیت مکانی
	استفاده از کتابخانه‌ی Modernizr جهت کنترل
262	پشتیبانی از قابلیت موقعیت مکانی
263	دریافت موقعیت حال حاضر کاربر

فصل هشتم: تغییر شکل‌ها و حرکت‌های انتقالی	
در CSS3 205	
205	تغییر شکل‌ها
206	حرکت یا همان translation
208	تغییر سایز یا همان scaling
210	چرخش
211	معوج‌سازی
212	جابه‌جا کردن مرکز تغییر
	پشتیبانی از اینترنت اکسپلورر 8 و نسخه‌های
212	قبل از آن
213	حرکت انتقالی
215	transition-property
216	transition-duration
217	transition-timing-function
218	transition-delay
219	خلاصه‌نویسی خصوصیات transition
	استفاده از چند جلوه‌ی انتقالی در یک نوبت
220	
221	انیمیشن‌ها
222	فریم‌های کلیدی
	خصوصیات مربوط به تکنیک‌های
223	متحرک‌سازی
223	animation-name
224	animation-duration
224	animation-timing-function
224	animation-iteration-count
224	animation-direction
225	animation-delay
225	animation-fill-mode
225	animation-play-state
	دستور خلاصه‌نویسی برای خصوصیات
225	انیمیشن
226	در ادامه چه خواهیم آموخت
فصل نهم: پیوست کردن فونت و ایجاد قالب	
چندستونه 227	
227	فونت بر بستر وب با استفاده از @font-face
229	استفاده از دستور @font-face

288	راه میان‌بر	263	آبجکت موقعیت مکانی position
289	حذف گزینه‌ها و پاک‌سازی داده‌ها	265	دریافت طول و عرض جغرافیایی
289	محدودیت در ذخیره‌سازی	266	بارگذاری نقشه
290	ملاحظات امنیتی	270	آخرین کلام برای ابزارهای همراه قدیمی
290	اضافه کردن قابلیت ذخیره‌سازی تحت وب به سایت HTML5 Herald	271	برنامه‌های تحت وب خارج از شبکه
296	APIهای دیگر HTML5	این قابلیت چطور کار می‌کند: مخزن برنامه‌های HTML5	
296	Web Workers	271	چطور سایت را برای فعالیت خارج از شبکه، تنظیم کنیم
297	Web Sockets	272	فایل cache.manifest
298	IndexedDB و Web SQL	273	تعیین نوع محتوا بر روی server
299	در ادامه چه یاد می‌گیریم؟	ارجاع دادن به فایل Manifest در سند HTML	
فصل یازدهم: Canvas, SVG و قابلیت Drag and Drop		275	کسب اجازه از کاربر برای ذخیره اطلاعات سایت برای استفاده‌ی آفلاین
301	Drop	276	آفلاین شوید و تست کنید
301	Canvas	تست جهت اطمینان از ذخیره شدن سایت در مخزن	
302	کمی درباره‌ی تاریخچه‌ی Canvas	277	اضافه کردن قابلیت دسترسی خارج از شبکه به سایت HTML5 Herald
302	ساخت عنصر canvas	278	محدود کردن حجم مخزن برنامه‌ی تحت وب
304	رسم بر روی canvas	279	بخش fallback
304	مشخص کردن محیط canvas	280	به روز کردن محتویات مخزن
انتخاب رنگ برای شروع به رسم بر روی canvas		281	در مخزن قرار گرفتن فایل cach.manifest
305	رسم یک چهار ضلعی بر روی canvas	282	ما هنوز آنلاین هستیم؟
306	سیستم مختصات در Canvas	283	منابع برای مطالعه بیشتر
307	سبک‌های مختلف رنگ‌آمیزی با fillStyle	284	ذخیره‌سازی تحت وب
311	رسم اشکال دیگر با ایجاد مسیر	285	دو شیوه‌ی مختلف ذخیره‌سازی تحت وب
ذخیره‌سازی اشکال ترسیم شده بر روی canvas		285	ذخیره‌سازی نشست
314	قرار دادن یک تصویر بر روی canvas	286	ذخیره‌سازی محلی
318	دستکاری تصاویر	داده‌های ذخیره شده توسط استاندارد ذخیره‌سازی تحت وب چه الگویی دارند؟	
319	تبدیل یک تصویر رنگی به سیاه و سفید	287	دریافت و ذخیره‌سازی داده‌ها
322	خطاهای امنیتی و متد getImageData	288	تبدیل داده‌های ذخیره شده
322	دستکاری فیلم ویدئویی بر روی canvas		
325	نمایش متن بر روی canvas		
329	نگرانی‌های مربوط به دسترسی‌پذیری		
329	منابع برای مطالعه بیشتر		
329	SVG		
330	رسم شکل با SVG		

347	ضمیمه الف: Modernizr	استفاده از Inkscape برای رسم تصاویر SVG
348	استفاده از Modernizr به همراه CSS	333
350	به‌کارگیری Modernizr با جاوااسکریپت	333 فیلترهای SVG
351	پشتیبانی از سبک‌دهی به برجسب‌های HTML در IE8 و نسخه‌های قدیمی‌تر	334 استفاده از کتابخانه‌ی رافائل
351	منابع برای مطالعه بیشتر	334 رسم یک تصویر با استفاده از رافائل
353	ضمیمه ی ب: WAI-ARIA	336 چرخاندن تصویر با استفاده از رافائل
354	چطور WAI-ARIA مکمل عناصر معناگرا می‌شود	337 Canvas در مقایسه با SVG
354	وضعیت کنونی WAI-ARIA	338 Drag and Drop (کشیدن و انداختن)
356	منابع برای مطالعه بیشتر	339 غذا دادن به گربه
357	ضمیمه ج: ریزداده	340 اضافه کردن قابلیت کشیدن شدن (draggable شدن) به عناصر
		341 آپجکت DataTransfer
		342 قبول کردن عنصرهای drop شده
		345 منابع برای مطالعه بیشتر
		345 دوستان به انتهای خط رسیدیم

درباره‌ی نویسنده‌گان

الکسیس گلدستین^۱

الکسیس گلدستین زمانی که در اواسط دهه نود در دبیرستان درس می‌خواند، HTML را فرا گرفت و سپس مدرک خود را در رشته علوم کامپیوتری از دانشگاه کلمبیا دریافت کرد. او صاحب شرکت aut faciam LLC است که در آن به توسعه نرم‌افزار و آموزش برنامه‌نویسی می‌پردازد. او قبل از اینکه شرکت خود را تأسیس کند، برای هفت سال در بخش فناوری در Wall Street مشغول به کار بود. او به عنوان مسئول سهام و برای سه شرکت بزرگ کار کرده است و آموخته است که از بررسی کدها لذت ببرد. او مدرس و یکی از سازمان‌دهندگان گروه Girl Develop It است. این گروه کلاس‌های برنامه‌نویسی ارزان قیمت برای بانوان برپا می‌کند. او یکی از اعضای پر افتخار گروه هکری NYC Resistor در بروکلین است. شما می‌توانید او را در وب سایتش در آدرس <http://alexisgo.com/> بیابید.

لوییس لازاریس^۲

لوییس لازاریس یک طراح آزاد و خود اشتغال است که به طراحی اینترنتی وب در تورنتوی کانادا مشغول است. او از سال 2000 در این صنعت فعالیت می‌کند. او از زمانی که در طراحی، جدول و تصاویر GIF یک پیسکی بسیار رایج بود، به طراحی مشغول است. در طول پنج سال گذشته او به سوی استفاده از استانداردهای وب روی آورده و سعی می‌کند تا بهترین شیوه‌های طراحی را تبلیغ کند که هم برنامه‌نویس و هم مشتری را به هدف می‌رساند. او به طور منظم برای برخی از وبلاگ‌های طراحی وب می‌نویسد که وبلاگ شخصی خودش نیز یکی از همین وبلاگ‌هاست. آدرس سایت او <http://impressivewebs.com> است.

استلا ویل^۳

استلا ویل؛ مهندس طراحی اینترنتی از سان‌فرانسیسکو است که از سال 1999 به طراحی سایت‌های استاندارد بر پایه اصول دسترسی‌پذیری اشتغال دارد. استلا کار بر روی CSS3 را در زمان ارائه آیفون در سال 2007 شروع کرد و بعد از چهار سال توسعه نرم‌افزار برای موبایل و مرورگر WebKit، تقریباً با تمامی اشکالات CSS3 در WebKit آشنا است و در به کارگیری اجزای HTML5 تجربه‌ی بالایی دارد. او برای دو وبلاگ فنی، مقالات آموزشی و مطالب مفصلی درباره‌ی پشتیبانی مرورگرها از CSS3 و HTML5 می‌نویسد. (<http://www.standardista.com/>) او به تدریس طراحی و توسعه‌ی وب علاقه‌مند است و در کنفرانس‌هایی در سراسر آمریکا درباره‌ی HTML5، CSS3، JavaScript و توسعه وب بر روی موبایل، صحبت می‌کند. (و او امید دارد که در سطح جهان این کار را انجام دهد.)

درباره بازبین فنی کتاب

1. Alexis Goldstein
2. Louis Lazaris
3. Stele weyl

راس ویکی برای مدت 18 سال است که در حیطه طراحی فعالیت می‌کند و تخصص اصلی او طراحی و توسعه وب و آموزش است. راس، یکی از دو رئیس گروه استانداردهای وب است و همچنین یکی از اعضا و بنیانگذاران سازمان متخصصان صنعت وب استرالیا (WIPA) به حساب می‌آید. راس، مجموعه‌ای از خودآموزهای CSS نوشته است که بسیار محبوبند و او را در سطح دنیا به خاطر سمینارها و کارگاه‌هایش می‌شناسند. او سایت Max Design (<http://mxdesign.com.au/>) را اداره می‌کند.

مقدمه

در سال‌های اخیر، گرایش به سمت طراحی استاندارد و استفاده از تکنولوژی‌های روز در بین طراحان افزایش پیدا کرده است و طراحان ایرانی نیز گرایش بسیاری به سمت استفاده از دانش روز دارند. در این بین، HTML5 و CSS3 تحولی عظیم را در طراحی وب به وجود آورده‌اند. این تکنولوژی‌ها، طراحی سایت را بسیار آسان کرده‌اند و اکنون می‌توانید بهتر از هر زمان دیگری، سایتی طراحی کنید که سبک، کارآمد و انعطاف‌پذیر باشد. همچنین ساخت برنامه‌های تحت وب و اپلیکیشن‌های موبایل نیز به مدد این تکنولوژی‌ها بسیار آسان شده است. اکنون با ابزارهایی همچون PhoneGap می‌توان با استفاده از HTML5، CSS3 و جاوااسکریپت، برنامه‌های پیشرفته و پیچیده برای پلتفرم‌های مختلف ایجاد کرد و برخی از برنامه‌هایی که هم اکنون بر روی گوشی تلفن همراه شما نصب شده است، ممکن است با استفاده از همین تکنولوژی‌ها ساخته شده باشد. حتی در Windows 8 تمهیداتی اندیشیده شده است که می‌توانید در این سیستم عامل با استفاده از HTML5، CSS3 و JavaScript برنامه‌هایی بنویسید که بر روی ویندوز اجرا شوند و عملاً به نظر خواهد رسید که این برنامه‌ها هیچ تفاوتی با برنامه‌های نوشته شده با زبان C یا Java ندارند. به عبارت دیگر، یادگیری این تکنولوژی در آینده نزدیک برای هر طراح، جزو ملزومات خواهد بود.

متأسفانه منابع بسیار اندکی در زمینه‌ی تکنولوژی‌های جدید و استانداردهای روز به زبان فارسی ترجمه می‌شود. با بررسی تعدادی از کتاب‌های مرتبط با این موضوع، این کتاب را برای ترجمه انتخاب کردم. شاید این کتاب مانند بسیاری از کتاب‌های خودآموز، مملو از مثال و نمونه کد نباشد، ولی با بررسی‌ای که انجام شد، این کتاب یکی از جامع‌ترین کتاب‌ها برای آموزش HTML5 و CSS3 است. این کتاب ماهیتی ما بین خودآموز و مرجع را دارد. از طرفی تمامی خصوصیات و قابلیت‌های HTML5 و CSS3 در این کتاب توضیح داده شده است و می‌توانید به عنوان یک مرجع از آن استفاده کنید و از طرفی برای هر کدام از این قابلیت‌ها، مثالی نیز ذکر شده است که کمک می‌کند، مفهوم آن قابلیت جدید را بهتر درک کنید.

بر اساس بازخوردهایی که از ترجمه کتاب‌های قبلی به دست آمد، به نظر می‌رسد زمانی که برای تمامی واژگان تخصصی، معادل فارسی در نظر گرفته شود، افراد متخصصی که قرار است از کتاب استفاده کنند، اندکی با مشکل مواجه می‌شوند، زیرا از طرفی با واژه‌های انگلیسی آشنا هستند و از سویی دیگر هیچ لغت واحدی به عنوان معادل برای واژه‌های تخصصی در نظر گرفته نشده است. برای حل این مشکل تصمیم گرفتم که هر جایی که قصد داشتم برای یک واژه تخصصی، معادل در نظر بگیرم، در پاورقی همان صفحه نیز واژه انگلیسی را ذکر کنم تا از این طریق هم کاربر بتواند با واژه جدید ارتباط برقرار کند و هم واژه اصل را بداند. در بخش‌های مختلف کتاب نیز هر جا که احساس کردم که مطلب روشن نیست و یا احتیاج به توضیحات اضافه دارد، بخش‌هایی را تحت عنوان «نکته از مترجم» آورده‌ام و سعی کردم توضیحات جامع‌تری را ارائه کرده و یا منابع دیگری را معرفی کنم.

امیدوارم با ترجمه این کتاب توانسته باشم اندکی به دوستان و همکارانی که به حرفه طراحی وب اشتغال دارند و یا به این حرفه علاقه‌مند هستند، کمکی کرده باشم.

معرفی گروه WeDesign

تیم طراحی WeDesign از سال 1388 کار خود را در زمینه‌ی طراحی وب سایت، آموزش طراحی وب و ترجمه کتاب‌ها و مقالات مرتبط با طراحی، آغاز کرد. هدف این تیم ارائه سرویس طراحی واسط کاربری وب، براساس آخرین استانداردهای روز، برگزاری کلاس‌های آموزشی و ترجمه کتاب‌های جدید و کاربردی طراحی وب است.

برای کسب اطلاعات بیشتر درباره‌ی فعالیت‌های این گروه، دوره‌های آموزشی و کتاب‌های ترجمه شده توسط اعضا، به سایت: <http://www.wedesign.ir/> مراجعه کنید.

از طریق آدرس ایمیل a.abdolali@gmail.com نیز می‌توانید با مترجم کتاب در تماس باشید.

محتویات DVD کتاب

به همراه کتاب، یک DVD نیز ارائه شده است که حاوی موارد زیر است:

- کدهای آماده‌ی پروژه‌های داخل کتاب
 - بیش از 10 ساعت فیلم آموزشی HTML5 و CSS3 به زبان انگلیسی
 - 40 فونت فارسی اصلاح شده در سه فرمت مختلف برای استفاده در وب
 - تعدادی کتاب مفید دیگر، برای کسانی که تمایل دارند دانسته‌های خود را ارتقا دهند.
- برای دیدن فیلم‌های آموزشی باید نرم‌افزار quicktime را بر روی سیستم خود نصب کنید. این نرم‌افزار نیز در DVD کتاب موجود است.

پیش‌گفتار

آیا نام یورد ویسچر را شنیده‌اید؟ احتمال می‌دهم که نشنیده باشید، اما اکتشاف کوچک او، استفاده از HTML5 را برایمان امکان‌پذیر کرد.

در سال 2002، در شهر هاگ در هلند، آقای ویسچر تصمیم داشت که سرعت اجرای خروجی فایل XSL خود را افزایش دهد. او به جای استفاده از دستور createElement در جاوا اسکریپت از دستور innerHTML استفاده کرد و متوجه شد که عناصر ناشناخته که بخشی از برچسب‌های HTML^۱ نیستند را نمی‌توان با استفاده از CSS، سبک‌دهی کرد.

در سال 2008، جنبش HTML5 قوت گرفت. عناصر تازه‌ای تعریف شدند، اما مرورگرهای اینترنت اکسپلورر نسخه 6 تا 8 به مشکل برخوردند و نمی‌توانستند عناصر جدید را شناسایی کنند؛ عناصر جدید در این مرورگرها نمی‌توانستند برچسبی را به عنوان فرزند در خود جای دهند و CSS نیز بر روی این برچسب‌های جدید اعمال نمی‌شد. همین حقیقت ناامیدکننده مانعی بر سر راه پیاده‌سازی HTML5 ایجاد کرد.

حال، یورد، نیم دهه بعد از اکتشافش ترفندی را در یکی از یادداشت‌هایش در کارگروه HTML^۲ کنسرسیوم جهانی وب در زیر مطلب سم روبی^۳ نوشت: «در هر صورت، اگر می‌خواهید که دستورات CSS به برچسب‌های ناشناخته در اینترنت اکسپلورر اعمال شود، تنها باید از دستور document.createElement(elementName) استفاده کنید و آن برچسب جدید را معرفی کنید. این ترفند به موتور پردازش CSS اطلاع می‌دهد که برچسبی با این نام هم وجود دارد و می‌توان به آن سبک اختصاص داد.»

ایان هیکسون، ویراستار ارشد مستندات HTML5، همانند همه فعالان وب از آموختن این ترفند، مسرور شد. او تا به حال از این ترفند اطلاع نداشت و با خوشحالی گفت: «این ترفند جدید کار را برای ساخت یک وصله ترمیمی برای IE7 بسیار آسان‌تر از آنی می‌کند که انتظارش را داشتیم.» یک روز بعد، جان رسیگ^۴ مطلبی در وبلاگش نوشت و از عبارت HTML shiv استفاده کرد. حال به اتفاقات بعدی دقت کنید:

- ژانویه 2009: رمی شارپ^۵ اولین اسکریپت برای فعال کردن HTML5 در مرورگر IE را منتشر کرد.
- ژوئن 2009: فاروق آتش دستورات HTML5 shiv را به نسخه اولیه کتابخانه Modernizer اضافه کرد.
- فوریه 2010: یک گروه از حرفه‌ای‌های جاوا اسکریپت از جمله رمی، کانگکس^۶، جان دیوید دالتون^۷، و پورنل^۸ با همکاری هم حجم اسکریپت اولیه را کاهش دادند.

۱. هر زمان که در متن از واژه عنصر HTML و یا برچسب HTML استفاده شد، منظور همان tag های HTML است. از آنجایی که در اغلب کتاب‌های ترجمه شده، واژه برچسب و عنصر را معادل tag گرفته‌اند، من نیز در این کتاب از همین اصول پیروی می‌کنم.

2. Working group
3. Sam Ruby
4. John Resig (این شخص مبدع کتابخانه jQuery نیز هست)
5. Remy Sharp
6. Kangax
7. John-David Dalton
8. porneL

- مارچ 2010: ماتیاس بیننز¹ و برخی دیگر از فعالان متوجه شدند که اسکریپت shiv تأثیری بر روی صفحه مناسب برای چاپ در IE نمی‌گذارد. روز غم‌انگیزی بود. من به طور غیررسمی از توسعه‌دهندگان وب خواستم که برای یافتن یک راه حل تازه، تلاش کنند.
 - آوریل 2010: جاناتان نیل² در پاسخ به درخواستم فایل IE Print Protector یا به اختصار IEPP را منتشر کرد که از همان دستورات HTML5 shiv استفاده می‌کرد، ولی در صفحه چاپ نیز پشتیبانی از دستورات HTML5 اضافه شده بود.
 - آوریل 2010: رمی دستورات قدیمی HTML5 shiv را با راه‌حل ارائه شده در IEPP تعویض کرد.
 - فوریه 2011: الکساندر فارکاس³ هدایت پروژه را به عهده گرفت و کد IEPP را به GitHub انتقال داد، تست‌های فراوانی برای آن ترتیب داد و ایراداتش را برطرف کرد و سرعتش را نیز افزایش داد.
 - آوریل 2011: IEPP v2 منتشر شد و کتابخانه Modernizr و HTML5 shiv از کدهای جدید ارائه شده در این نسخه استفاده کردند و حال همگان بدون نگرانی در حال استفاده از دستورات HTML5 در طراحی‌هایشان هستند.
- داستان HTML5 shiv تنها یک نمونه از کارهایی است که نشان‌دهنده همکاری افراد مختلف برای پیشبرد وب آزاد است. این تنها کنسرسیوم جهانی وب و یا مرورگرها نیستند که بر روی استفاده ما از وب تأثیر می‌گذارند، بلکه افرادی همانند شما و ما نیز در این مسیر نقش داریم. امیدوارم این کتاب شما را تشویق کند تا به همین شکل عمل کنید؛ بهترین راه برای کسب مهارت بیشتر این است که دانسته‌هایتان را فعالانه با دیگران به اشتراک بگذارید.
- به کارگیری HTML5 و CSS3 در حال حاضر از هر زمان دیگری آسان‌تر و مفرح‌تر است. این کتاب مطالب کاربردی و باارزش زیادی را در اختیارتان قرار می‌دهد و چیزهایی را که باید بدانید، به شما می‌آموزد تا از HTML5 بهره ببرید. نویسندگان - الکسیس، لوییس و استلا - توسعه‌دهندگان مطرحی هستند که شیوه آموزش‌شان، واقعی است و به شما کمک می‌کند تا بهترین شیوه‌های استفاده از HTML5 را به آسانی بیاموزید.
- من ایمان دارم که این کتاب می‌تواند برایتان مفید باشد، و شما نیز مانند من از نسل بعدی وب، هیجان‌زده خواهید شد.

پول آیریش⁴

رابط توسعه‌دهندگان jQuery

توسعه‌دهنده ارشد کتابخانه Modernizr و HTML5 Boilerplate

آوریل 2011

-
1. Mathias Bynens
 2. Janathan Neal
 3. Alexander Farkas
 4. Paul Irish

مقدمه کتاب

از اینکه خودآموز HTML5 و CSS3 را انتخاب کردید از شما متشکریم. از اینکه تصمیم گرفتید در این سفر برای اکتشاف جدیدترین و بزرگترین تکنولوژی‌های طراحی با ما همراه شوید، بسیار خوشحالیم. اگر یک نسخه از این کتاب را در دست دارید، احتمال می‌دهیم که حداقل در سطح ابتدایی با HTML و CSS آشنایی دارید. حتی ممکن است در زمینه‌ای خاص از کدنویسی، سبک‌دهی و اسکریپت‌نویسی، تخصص داشته باشید و حال می‌خواهید که مهارت خود را توسعه دهید و ویژگی‌های جدید مربوط به HTML5 و CSS3 را بیاموزید.

یادگرفتن موضوعی جدید می‌تواند سخت باشد. ممکن است وقت‌تان محدود باشد و نتوانید مستندات رسمی مربوط به هر کدام از این زبان‌های تحت وب را به طور کامل بخوانید. ممکن است خواندن یکی از کتاب‌های بیش از حد فنی خسته‌تان کرده باشد، زیرا این کتاب‌ها اغلب مرجع هستند و مثال‌های واقعی و کاربردی کمی در خود دارند.

در نهایت، هدف ما از نوشتن این کتاب این بوده که به شما کمک کنیم تا از طریق دستورالعمل‌های کاربردی و با ذکر مثال بتوانید با مشکلاتی که در واقعیت در حین طراحی با آنها برخورد می‌کنید، روبه‌رو شوید و در همه این مراحل تأکید ما بر روی HTML5 و CSS3 است.

اما این کتاب چیزی بیش از یک خودآموز است. در طول کتاب، اطلاعات تئوری و فنی بسیاری به شما ارائه خواهد شد و کمکتان می‌کنیم تا شکاف موجود درباره درک این تکنولوژی‌ها (و اینکه چطور کار می‌کنند و چرا ایجاد شده‌اند) را پر کنید، در حالی که نهایت سعی‌مان را می‌کنیم که با داده‌های جدید و زیاد، کلافه‌تان نکنیم. اجازه دهید شروع کنیم!

این کتاب برای چه کسانی مناسب است؟

این کتاب برای توسعه‌دهندگان و طراحان وبی نوشته شده است که می‌خواهند جدیدترین فناوری‌های مبتنی بر مرورگر را فرا بگیرند. ممکن است دانش شما در HTML و CSS در سطح متوسط باشد، از این رو ما زمانی را صرف توضیح مقدمات کدنویسی HTML و سبک‌دهی با CSS نخواهیم کرد. در عوض، به شما می‌آموزیم که چه امکاناتی در قالب HTML5 و CSS3 در اختیارتان قرار دارد.

نکته از مترجم: این کتاب برای افرادیست که با HTML و CSS آشنایی دارند. در صورتی که با HTML و CSS آشنایی ندارید. به شما پیشنهاد می‌کنم که کتاب‌های مقدماتی‌تر در این زمینه را مطالعه کنید و سپس این کتاب را بخوانید. برای یادگیری HTML می‌توانید از کتاب خودآموز HTML نوشته‌ی الیزابت کسترو و ترجمه‌ی آقای میرمحمدجواد معتمد‌هاشمی استفاده کنید که توسط انتشارات ناقوس به چاپ رسیده است. برای فراگیری CSS نیز می‌توانید از کتاب «101 نکته و ترفند برای استفاده از CSS در طراحی وب» بهره ببرید که توسط اینجانب به فارسی ترجمه شده و توسط انتشارات ناقوس نیز به چاپ رسیده است.

دو فصل آخر این کتاب به بررسی برخی از API های جدید جاوااسکریپت می پردازد که به HTML5 مربوط است. برای درک این فصل ها لازم است که مقدمات جاوااسکریپت را بدانید، اما دانستن جاوااسکریپت برای فهمیدن بقیه بخش های کتاب ضروری نیست. اگر با جاوااسکریپت آشنایی ندارید، می توانید بدون مشکل از خواندن این فصل ها صرف نظر کنید و زمانی که با جاوااسکریپت آشنایی پیدا کردید، بازگردید و آنها را بخوانید.

این کتاب حاوی چه مطالبی است؟

این کتاب شامل یازده فصل و سه ضمیمه است. اغلب فصل ها بر ادامه فصل قبل بنا شده اند، از این رو اگر آنها را به ترتیب بخوانید، بهترین نتیجه را می گیرید، اما اگر می خواهید تنها یک موضوع مشخص را بخوانید و اطلاعاتتان را تازه کنید، می توانید فصل ها را جا بیاندازید.

فصل اول: معرفی HTML5 و CSS3

قبل از اینکه به مطالب کاربردی برسیم، مقداری از تاریخچه این زبان های نشانه گذاری¹ برایتان تعریف می کنیم و همچنین دلایلی قانع کننده را ذکر می کنیم تا از همین امروز از HTML5 و CSS3 استفاده کنید. سپس به وضعیت پشتیبانی مرورگرها از این زبان ها می پردازیم و توضیح می دهیم که اغلب این فناوری های جدید، امروزه قابل استفاده اند - البته به شرطی که به درستی به کار گرفته شوند.

فصل دوم: نشانه گذاری، سبک های HTML5

در این فصل، به شما برخی از عناصر معنایی و ساختاری جدید را که به HTML5 اضافه شده است را معرفی می کنیم. سپس سایت HTML5 Herald را به شما معرفی می کنیم که در اصل یک سایت نمایشی است. ما در طول کتاب بر روی آن کار می کنیم. فکر می کنید برچسب های div خسته کننده شده اند؟ ما هم همینطور فکر می کنیم. گزینه های خوبی که HTML5 در حال حاضر در اختیارمان قرار می دهد، متعددند. برچسب هایی همچون: header, article, section, nav, footer, aside!

فصل سوم: برچسب های معناگرای² بیشتری از HTML5

در ادامه مطالب فصل قبل، توجهمان را به راه جدیدی معطوف می کنیم که HTML5 از طریق آن رؤس مطالب را منظم می کند. سپس به بررسی عناصر متعدد و معناگرای دیگری می پردازیم که به شما اجازه می دهند تا کدی پرمعنا تر بنویسید.

فصل چهارم: فرم های HTML5

یکی از سودمندترین قابلیت های HTML5 که در حال حاضر قابل استفاده است، به فرم ها مربوط می شود. برخی از مرورگرها در حال حاضر به شکل ذاتی، صحت اطلاعات وارد شده در فیلدهای ایمیل و یا آدرس صفحات وب (URL) را چک می کنند و برخی دیگر از مرورگرها به شکل ذاتی از انتخاب کننده های تاریخ، لغزنده ها³ و جعبه های چرخشی⁴ پشتیبانی می کنند. همین ها کفایت تا از طراحی

-
1. Markup language
 2. semantic
 3. Date picker
 4. slider
 5. Spinner box

فرم‌ها لذت ببرید! این فصل تمامی اطلاعاتی را که برای سرعت‌بخشی در کدنویسی فرم‌ها با HTML5 نیاز دارید را در اختیارتان قرار می‌دهد و همچنین کدهایی نیز نوشته می‌شود که در صورت عدم پشتیبانی مرورگر از این قابلیت‌ها، مرورگرهای قدیمی از این کدها استفاده کنند و به عبارتی با استفاده از جاوااسکریپت همین ویژگی‌ها برای مرورگرهای قدیمی شبیه‌سازی می‌شود.

فصل پنجم: صوت و تصویر در HTML5

از HTML5 به عنوان رقیبی یاد می‌شود که آمده تا بر سیطره‌ی سلطان چندرسانه‌ای اینترنت یعنی فرمت فلش، خاتمه دهد. عناصر audio و video ابزارهایی هستند که قرار است این کار را انجام دهند – آنها به شکل ذاتی بستری را فراهم می‌کنند که هم قابل برنامه‌نویسی شدن با جاوااسکریپت را داراست و هم به افزونه‌های شخص ثالثی همچون فلش احتیاجی ندارد. در این فصل همه نکات لازم برای استفاده از این برچسب‌های تازه را فرا می‌گیرید.

فصل ششم: معرفی CSS3

حال که همه چیز را درباره‌ی HTML5 توضیح دادیم، زمان آن رسیده تا به بررسی زبان دوست و نزدیک دیگری؛ به اسم CSS3 بپردازیم. ما سفر خود را با بررسی برخی از گزینش‌گرهای^۱ جدید آغاز می‌کنیم که به شما اجازه می‌دهد تا با انعطافی بی‌سابقه، عناصر داخل صفحه را انتخاب کنید. سپس به شیوه‌های جدید تعریف رنگ‌ها در CSS3 می‌پردازیم که شفافیت را نیز شامل می‌شود. سپس فصل را با ارائه ترفند سریع و مفید (قابلیت‌های جذاب CSS3 که با اندک زحمتی می‌توانید به سایت خود اضافه‌شان کنید مانند، سایه‌ی متن، سایه‌ی جعبه و گوشه‌های گرد) به پایان می‌رسانیم.

فصل هفتم: توالی رنگ^۲ و استفاده از چند تصویر پس‌زمینه در CSS3

آخرین باری که سایتی را طراحی کردید که در طرحش توالی رنگ و تصاویر پس‌زمینه وجود نداشت را به یاد می‌آورید؟ CSS3 برای این دو مورد امکاناتی را ارائه می‌کند. البته این ویژگی‌ها خیلی دیر در اختیار طراحان قرار گرفته است. در گذشته، طراحان زمان زیادی را صرف ساخت تصاویر پس‌زمینه و یا توالی رنگ‌ها می‌کردند. از سویی باید به بی‌نقص بودن آن فکر می‌کردند و از سوی دیگر به پایین‌نگه داشتن حجم تصاویر تولید شده بودند. حال شما می‌توانید توالی رنگ‌های خطی و دایره‌ای را مستقیماً در داخل CSS تعریف کنید و استفاده از تصاویر را کنار بگذارید، همچنین می‌توانید هر تعداد که می‌خواهید، برای یک عنصر، تصویر پس‌زمینه در نظر بگیرید. زمان آن رسیده که دیگر برچسب‌های div تودرتو را که برای قرار دادن چند تصویر پس‌زمینه از آنها استفاده می‌کردید را حذف کنید.

1. selector

2. gradient

فصل هشتم: جلوه‌های تغییر شکل^۱ و جلوه‌های انتقالی در CSS3

پویانمایی یا همان انیمیشن از مدت‌ها قبل جزو قلمرو جاوااسکریپت به حساب می‌آمد، اما CSS3 به شما اجازه می‌دهد که تولید برخی از این جلوه‌ها را به مرورگر واگذار کنید. جلوه‌های تغییر شکل به شما اجازه می‌دهد تا عنصری را در طرح بچرخانید، معکوس کنید، معوج کنید و یا حتی عنصری را مخفی کنید. این جلوه‌های تغییر شکل می‌تواند بسیار ساده و یا بسیار پیچیده باشد. این فصل را با نگاهی به آینده به اتمام می‌رسانیم؛ درست است که در حال حاضر انیمیشن‌های مبتنی بر keyframe در حال حاضر به طول کامل توسط مرورگرها پشتیبانی نمی‌شود، اما فکر می‌کنیم شما هم مانند ما معتقدید که این انیمیشن‌ها بسیار دوست‌داشتنی هستند.

فصل نهم: پیوست‌کردن فونت و طرح‌های چند ستونه

فونت Arial را ترجیح می‌دهید یا فونت Verdana را؟ Georgia یا Times؟ شاید هم هیچکدام‌شان؟ در این فصل، نگاهی می‌اندازیم به شیوه‌هایی که از طریق آنها می‌توان از فونت‌های امن و رایج وب فراتر رفته و هر فونتی را که مایلیم به صفحه پیوست کرده تا در زمان بارگذاری^۲ تصاویر و برگه سبکتان، فونت موردنظرتان هم بر روی سیستم کاربر بارگذاری شود. سپس به ویژگی جدیدی از CSS نگاهی می‌اندازیم که با استفاده از آن می‌توانیم محتوای صفحاتمان را به شکل چندستونه نمایش دهیم، بدون اینکه لازم باشد برچسبی را به صفحه اضافه کرده و یا از دستور float استفاده کنیم.

فصل دهم: موقعیت مکانی^۳، برنامه‌های خارج از شبکه^۴ و ذخیره‌سازی تحت وب^۵

مرورگرهای نسل جدید به APIهای جاوااسکریپت مختلفی مجهز شده‌اند. بسیاری از این ویژگی‌ها برای مرورگرهای موبایل طراحی شده‌اند، اما می‌توانند برای کاربران معمولی وب هم مفید باشند. در این فصل به سه مورد از هیجان‌انگیزترین این امکانات نگاهی می‌اندازیم: موقعیت مکانی، برنامه‌های خارج از شبکه و ذخیره‌سازی تحت وب. همچنین برخی از این APIها را تنها به طور مختصر معرفی می‌کنیم و به طور کامل توضیح‌شان نمی‌دهیم، زیرا اغلب پشتیبانی ضعیفی از آنها صورت گرفته و یا مورد مصرف کمتری دارند – همچنین منابعی را به شما معرفی می‌کنیم که می‌توانید از آنها برای کسب اطلاعات بیشتر درباره این APIها استفاده کنید.

فصل یازدهم: Drag and Drop و Canvas, SVG

نخست، فصل اول کتاب را به دو فناوری اختصاص می‌دهیم که برای رسم و نمایش تصویر ایجاد شده‌اند و البته این دو فناوری رقیب یکدیگر محسوب می‌شوند. Canvas به تازگی به HTML5 اضافه شده است و سطح پیکسلی را فراهم می‌کند و دارای API جاوااسکریپت خاص خود است که به شما اجازه می‌دهد تا با استفاده از آن، تصاویر موردنظرتان را رسم کنید. از سوی دیگر SVG سال‌هاست که وجود

-
1. transform
 2. download
 3. Geolocation
 4. Offline Web App
 5. Web Storage

دارد ولی به تازگی پشتیبانی خوبی از آن در مرورگرها صورت گرفته است، از این رو جایگزین مناسبی برای مورد قبلی محسوب می‌شود. در آخر، به بررسی API جاوااسکریپت جدیدی می‌پردازیم - Drag and Drop یا همان کشیدن و رها کردن - که به طور ذاتی قابلیت کشیدن و رها کردن عناصر در صفحه را در مرورگر ممکن می‌کند.

ضمیمه الف: Modernizr

یک ابزار کلیدی در میان ابزارهای موجود برای HTML5 همین ابزار است. Modernizr کتابخانه جاوااسکریپت آراسته و کوچکی است که تقریباً پشتیبانی از تمامی امکانات HTML5 و CSS3 را شناسایی می‌کند و به شما اجازه می‌دهد که سبک‌های CSS خود را به شکل گزینشی بنویسید. در این بخش ما یک خودآموز سریع برای استفاده از Modernizr آورده‌ایم. از این کتابخانه در طول کتاب هم استفاده شده است. به این شکل شما یک مرجع آماده را در پایان کتاب در اختیار دارید، در حالی که در فصل‌های دیگر بر روی خود HTML5 و CSS3 تمرکز شده است.

ضمیمه ب: WAI-ARIA

این استاندارد، مستندات مربوط به خودش را دارد و گاهی به اندازه HTML5 از آن نام برده می‌شود. WAI-ARIA مجموعه‌ای از آخرین ابزارهای موجود هستند که به شما کمک می‌کنند تا یک برنامه تحت وب پیچیده بسازید، به طوری که این برنامه برای کاربران ناتوان جسمی هم در دسترس باشد و فناوری‌های ایجاد شده برای افراد ناتوان جسمی بتوانند با برنامه شما ارتباط برقرار کنند. برای مبحثی همچون WAI-ARIA می‌توان یک کتاب کامل نوشت، ولی ما فکر می‌کنیم که اضافه کردن یک خلاصه مختصر و کوتاه در مورد این فناوری می‌تواند مفید باشد. علاوه بر آن منابعی را ذکر می‌کنیم تا در صورتی که به مبحث علاقه‌مند شده‌اید، بتوانید درباره آن بیشتر بیاموزید.

ضمیمه ج: ریزداده¹

ریزداده‌ها بخشی از مستندات HTML5 محسوب می‌شوند که مربوط به اضافه کردن یک سری label به برخی از برچسب‌های HTML است. این labelها تنها برای ماشین قابل خواندن هستند. البته فعالیت بر روی مستندات ریزداده‌ها بسیار زیاد است و اطلاعات مربوط به آن به سرعت تغییر می‌کند، اما فکر می‌کنیم که آوردن چند مثال برای آشنا کردن شما با این مقوله خالی از لطف نباشد.

عرف‌های رعایت شده در این کتاب

احتمالاً متوجه شده‌اید که ما در این کتاب از سبک خاصی برای صفحه‌آرایی استفاده کرده‌ایم تا تفاوت بخش‌های مختلف متن برای شما آشکار شود. به این موارد دقت کنید:

1. Microdata

نمونه کدها

کدها در این کتاب به این شکل به نمایش درمی آیند:

```
<h1>A Perfect Summer's Day</h1>
<p>It was a lovely day for a walk in the park. The birds were singing and the kids were all back at school.</p>
```

اگر کد در میان آرشیو کدهای کتاب باشد، نام فایلی که این کد مربوط به آن است، در بالا کد آورده می شود. همانند این:

example.css

```
.footer {
  background-color: #CCC;
  border-top: 1px solid #333;
}
```

اگر تنها بخشی از فایل نمایش داده شود، از کلمه `excerpt` در کنار نام فایل استفاده شده است:

example.css (excerpt)

```
border-top: 1px solid #333;
```

اگر تکه کدی به میان کدهای قبلی اضافه شود، بخش جدید را با فونت ضخیم به نمایش در می آوریم:

```
function animate() {
  new_variable = "Hello";
}
```

اگر بخواهیم یک تکه کد را تکرار کنیم و تنها یک بخش کوچک از کد مورد نیاز باشد، به جای بقیه ی کد از علامت سه نقطه عمودی استفاده می کنیم:

```
function animate() {
  :
  return new_variable;
}
```

نکته‌ها، ترفندها و هشدارها

در این بخش اطلاعات کوتاه و مفیدی نوشته می‌شود.



اطلاعات داخل این بخش مفید است، اما دانستن آنها ضروری نیست. به آنها به چشم داده‌های تکمیلی نگاه کنید.



معمولاً اطلاعات مهم را در این بخش می‌نویسیم.



مواردی که می‌تواند برایتان مشکل‌ساز باشد را در این بخش می‌نویسیم.

فصل اول

معرفی HTML5 و CSS3

این فصل در مورد اینکه چطور این استانداردها به این جایی که اکنون هستیم، رسیده‌اند و اینکه چرا HTML5 و CSS3 برای وب سایت‌های امروزی و نرم‌افزارهای تحت وب حائز اهمیت‌اند، توضیحات مختصری می‌دهیم. همچنین توضیح می‌دهیم که این فناوری‌ها چقدر برای آینده‌ی کاری شما به عنوان یک طراح وب، ارزشمند هستند.

اگر بخواهید تنها به اصل ماجرا بپردازید و پروژه‌هایی را که در کتاب آورده‌ایم، ببینید و شروع به یادگیری کنید و همچنین استفاده از قابلیت‌های جدید HTML5 و CSS3 را فرا بگیرید، می‌توانید این فصل را نادیده گرفته و به فصل دوم بروید و هر زمانی که به مطالب این فصل احتیاج داشتید، برگردید و این فصل را هم مطالعه کنید.

HTML5 چیست؟

چیزی که امروزه از آن به عنوان HTML5 یاد می‌شود، تاریخچه پر فراز و نشیبی دارد. به احتمال بسیار زیاد می‌دانید که زبان نشانه‌گذاری HTML در حال حاضر زبان غالب بر روی اینترنت است و از آن برای توصیف محتوا یا داده بر روی وب استفاده می‌شود. HTML5 آخرین نسخه از این زبان نشانه‌گذاری^۱ است که حاوی امکانات جدیدی است، برخی از امکاناتی که در نسخه قبلی موجود بود در این نسخه بهبود یافته و همچنین API‌های اسکریپت‌نویسی جدیدی نیز به آن اضافه شده است.

گفته می‌شود که HTML5 ویرایش تغییر یافته‌ی نسخه‌ی قبلی محسوب نمی‌شود - این نسخه شامل تمامی عناصر HTML4 و XHTML1.0 است. علاوه بر آن، در طراحی این نسخه اصول پایه‌ای مدنظر قرار گرفته تا اطمینان حاصل شود که این نسخه بر روی تمامی پلتفرم‌ها به خوبی کار خواهد کرد، با مرورگرهای قدیمی سازگار است و خطاهای برنامه‌نویسی را نیز به خوبی و بدون مشکل مدیریت

1. Markup Language

می‌کند. خلاصه‌ای از اصول طراحی که منجر به خلق HTML5 شده است را می‌توانید در صفحه اصول طراحی HTML در سایت W3C بخوانید^۱.

اول از همه، HTML5 تعریفی جدید برای عناصر نشانه‌گذاری موجود ارائه می‌دهد و عناصر جدیدی نیز به این نسخه اضافه شده است که به طراح اجازه می‌دهد تا کدی بامعنا تر بنویسد. چه لزومی دارد وقتی می‌توانید از برچسب‌های بامعنی‌تری همچون article, section, header, footer و بسیاری از برچسب‌های دیگر استفاده کنید، صفحه‌تان را با برچسب‌های div پر کنید؟

علاوه بر این، واژه‌ی HTML5 را برای تعدادی دیگر از فناوری‌ها و API‌های جدید دیگر نیز به کار می‌برند. برخی از این فناوری‌ها عبارتند از رسم تصاویر با استفاده از برچسب <canvas>، ذخیره‌سازی خارج از شبکه، برچسب‌های <video> و <audio>، قابلیت کشیدن و رها کردن عناصر داخل صفحه، پیوست کردن فونت به صفحه و دیگر قابلیت‌ها. در این کتاب، برخی از این فناوری‌ها را توضیح می‌دهیم.

API چیست ؟

API مخفف عبارت محیط برنامه‌نویسی نرم‌افزار است^۲. API را محیطی شبیه به محیط گرافیکی واسط کاربری^۳ تصور کنید - با این تفاوت که به جای اینکه این واسط کاربری برای انسان ساخته شده باشد، محیطی برای کدنویسی فراهم کرده است. API، مجموعه‌ای از دکمه‌ها (توابع از پیش نوشته شده) را برای کد شما فراهم می‌کند که با فشردن هر کدام از آنها، رویداد خاص موردنظر شما در سیستم، کتابخانه‌ی نرم‌افزار و یا مرورگر، فعال خواهد شد.

استفاده از دستورات بر پایه API راهی است برای خلاصه کردن دستورات پیچیده‌ای که قرار است در پس‌زمینه اجرا شوند. (یا گاهی اوقات توسط نرم‌افزار جانبی انجام شوند). برخی از API‌ها مرتبط با HTML5 در بخش‌های بعدی کتاب، معرفی و بررسی خواهند شد.

در مجموع، اگر تجربه‌ای در اسکریپت‌نویسی با API‌های جاوااسکریپت ندارید، نباید وحشت‌زده شوید. اگرچه تجربه داشتن در کدنویسی جاوااسکریپت بسیار سودمند است، ولی دانستن آن ضروری نیست.

هر زمانی که لازم باشد، بخش‌های مربوط به اسکریپت‌نویسی را به مرور در طول کتاب در اختیارتان قرار می‌دهیم تا مطمئن شویم این بخش‌ها را درک خواهید کرد.

لازم است که اشاره کنیم، برخی از فناوری‌هایی که زمانی بخشی از مستندات HTML5 بودند، اکنون جدا شده‌اند، از این رو از لحاظ فنی دیگر بخشی از HTML5 به حساب نمی‌آیند. برخی از فناوری‌های به خصوص دیگر نیز هرگز بخشی از HTML5 نبوده‌اند، این فناوری‌ها نیز گاهی بخشی از HTML5 در نظر گرفته می‌شوند. از این رو مجبور می‌شوید از واژه‌هایی با مفاهیم گسترده‌تر استفاده

1. <http://www.w3.org/TR/html-design-principles/>

2. Application Programming Interface

3. User interface

کنید مانند «HTML5 و فناوری‌های مرتبط با آن» بروس لاوسن^۱ به شوخی عبارت NEWT^۲ را به عنوان واژه معادل به کار برد که مخفف عبارت فناوری‌های جدید و هیجان‌انگیز وب^۳ است. به هر حال، جهت خلاصه نگاه‌داشتن کتاب و همچنین جلوگیری از بروز بحث‌های شدید بر سر اسم، ما به طور عام به تمامی این فناوری‌ها، HTML5 می‌گوییم.

چطور به اینجا رسیدیم؟

وب در بازه زمانی اندکی رشد کرده است. دوازده سال قبل، سایتی که از تصویر و عناصر چشم‌نواز استفاده می‌کرد، سایت به روز و برجسته‌ای به حساب می‌آمد.

اکنون، چهره وب تغییر کرده است. نرم‌افزارهای تحت وب ساده، سریع که بر پایه متدهای Ajax^۴ ساخته شده‌اند و از طرفی به کدنویسی طرف کاربر^۵ وابسته‌اند، هر روز بیشتر از قبل مورد استقبال قرار می‌گیرند. وب سایت‌های امروز، شبیه به نرم‌افزارهای کامپیوترهای رومیزی عمل می‌کنند و بسیاری از توسعه‌دهندگان وب نیز به سمت طراحی وب سایت با این اصول، پیش می‌روند.

در طول این مسیر، زبان نشانه‌گذاری وب نیز متحول شده است. HTML4 در نهایت جای خود را به XHTML داد که در اصل همان دستورات HTML4 را در خود داشت، ولی از اصول کدنویسی سخت‌گیرانه‌تر XML استفاده می‌کرد. در حال حاضر در دو نسخه HTML4 و XHTML بر روی وب به طور گسترده استفاده می‌شوند، اما HTML5 نیز در حال محبوب‌شدن است.

HTML5 در آغاز با دو نسخه سند مختلف ایجاد شد: Web Forms 2.0 و Web Apps 1.0 که هر دوی این سندها به خاطر تغییر ماهیت وب و نیاز به داشتن نرم‌افزارهای تحت وب سریع‌تر، کارآمدتر و قابل نگهداری‌تر، ایجاد شدند. فرم‌ها و قابلیت‌هایشان که شبیه به نرم‌افزارهای کامپیوترهای رومیزی است، قلب نرم‌افزارهای تحت وب به حساب می‌آیند، از این رو سمت و سوی حرکت HTML5 نیز در راستای همین خواسته‌هاست. در نهایت، این دو سند با هم ادغام شدند تا چیزی ایجاد شود که ما در حال حاضر آن را با نام HTML5 می‌شناسیم.

در زمانی که HTML5 در حال توسعه بود، بر روی XHTML 2.0 نیز کار می‌شد. پروژه‌ی توسعه XHTML 2.0 متوقف شده است تا تنها بر روی توسعه HTML5 تمرکز شود.

می‌شود مشخص کنید که کدام نسخه از مستندات HTML5 واقعی است؟

از آنجایی که مستندات HTML5 توسط دو گروه مختلف تدوین می‌شود (کنسرسیوم جهانی وب یا همان W3C و همچنین WHATWG)، دو نسخه مختلف از این سند موجود است. W3C (کنسرسیوم جهانی وب)

1. Bruce Lawson

2. <http://www.brucelawson.co.uk/2010/meet-newt-new-exciting-web-technologies/>

3. New Exciting Web Technologies

4. تلفظ صحیح انگلیسی این واژه «ای جکس» است.

5. Client-side

که احتمالاً برایتان آشناست: سازمانی است که مستندات اصلی HTML و CSS را نگهداری می‌کند و همچنین میزبان بسیاری از استانداردهای مرتبط با وب همچون SVG (گرافیک برداری مقیاس‌پذیر^۱) و WCAG (راهنمای دسترسی‌پذیری^۲ محتوای وب)

از سوی دیگر شاید WHATWG (یا همان کارگروه فناوری برنامه‌های ابرمتن تحت وب^۳) برایتان جدید باشد. این گروه توسط افرادی از شرکت‌های Apple, Mozilla و Opera تشکیل شده است که در سال 2004 کنسرسیوم جهانی وب درخواست‌شان را برای توسعه HTML رد کرد. آنها احساس می‌کردند که W3C نیاز کاربران و شرکت‌های تولیدکننده مرورگر را نادیده گرفته و به جای تمرکز بر روی استانداردهای HTML که بر روی مرورگرهای قدیمی هم قابل استفاده است، بر روی XHTML 2.0 تمرکز کرده. از این رو این گروه راه خود را در پیش گرفت و مستندات برنامه‌های تحت وب و فرم‌های تحت وب که بحث‌شان قبلاً مطرح شد را تهیه کردند. این دو سند با هم ادغام شدند و نام HTML5 بر آن نهاده شد. بعد از این اتفاق، W3C مستندات فراهم شده توسط این گروه را دریافت کرد و مستندات دیگری را براساس همین سند گروه WHATWG تهیه کرد و مستندات مربوط به HTML5 مخصوص به خودش را توسعه داد.

شاید کمی گیج‌کننده باشد. درست است، سیاست‌های پشت پرده وجود دارد که ما به عنوان طراح و برنامه‌نویس، کنترلی بر آنها نداریم. ولی آیا وجود دو سند متفاوت می‌تواند باعث نگرانی‌مان شود؟ در یک کلام، خیر.

مستندات تهیه شده توسط WHATWG را می‌توانید در آدرس <http://www.whatwg.org/html/> بیابید که به تازگی نامش به HTML تغییر کرده است. (عدد 5 از نام سند حذف شده است.) در حال حاضر به این سند، استاندارد زنده هم گفته می‌شود، زیرا به‌طور دائم در حال توسعه و تغییر است و دیگر نمی‌توان با شماره نسخه به آن اشاره کرد^۴.

نسخه تهیه شده توسط WHATWG تنها حاوی اطلاعات و قابلیت‌هایی است که تنها در HTML وجود دارد و امکانات جدید HTML5 نیز در این سند گنجانده شده است. علاوه بر آن، مستندات دیگری توسط WHATWG تهیه شده است که فناوری‌های مرتبط دیگر را شامل می‌شود. این مستندات موارد همچون ریزداده‌ها^۵، محتوای Canvas دوبعدی، Web Workerها، Web Storageها و برخی موارد دیگر را شامل می‌شود^۶.

1. Scalable vector graphic

2. accessibility

3. Web Hypertext Application Technology Working Group

4. برای کسب اطلاعات بیشتر درباره‌ی دلیل این تغییر نام به این آدرس مراجعه کنید.

<http://blog.whatwg.org/html-is-the-new-html5/>

5. Microdata

6. برای کسب اطلاعات بیشتر به این آدرس مراجعه کنید.

http://wiki.whatwg.org/wiki/FAQ#What_are_the_various_versions_of_the_spec.3F

مستندات تهیه شده توسط W3C را می‌توانید در آدرس <http://dev.w3.org/html5/spec/> ببینید. مستندات مربوط به دیگر فناوری‌ها نیز از آدرس <http://dev.w3.org/html5/> در دسترس است. حال تفاوت بین مستندات W3C و WHATWG در چیست؟ به طول خلاصه، مستندات WHATWG کمی محاوره‌ای‌تر و کاربردی‌تر است. (و البته بعضی‌ها می‌گویند کمی آینده‌گرانه‌تر است.) ولی در مجموع، این دو سند بسیار به هم شبیه هستند و می‌توان هر کدام از این دو سند را به عنوان منبعی برای یادگیری عناصر HTML5 و فناوری‌های مرتبط با آن به کار گرفت.

چرا HTML5 باید برایم مهم باشد؟

همانطور که قبلاً توضیح داده شد، هسته HTML5 تشکیل شده است از تعدادی عنصر معناگرا^۱ به همراه تعدادی فناوری و API‌های مرتبط. این تغییرات و عناصر جدید بدین منظور ایجاد شده‌اند تا کدنویسی یک صفحه وب، دسترسی و استفاده از آن را آسان‌تر کنند.

عناصر معناگرای جدید به همراه استانداردهایی همچون WAI-ARIA و Microdata (که درباره‌شان در ضمیمه B و C صحبت خواهیم کرد)، به ما کمک می‌کنند تا صفحاتی در دسترس‌تر برای انسان و ماشین فراهم کنیم – نفع این کار دو سویه است و هم به دسترس‌پذیر بودن صفحه ما کمک می‌کند و هم موتورهای جستجو بهتر می‌توانند مطالب داخل صفحه‌ی ما را کاوش کنند.

عناصر معناگرا نیز به طور خاص برای وب پویا طراحی شده‌اند و تمرکزشان بر روی این موضوع است که صفحات ماهیتی ماژولار پیدا کنند و استفاده از کدهای نوشته شده در بخش‌های دیگر نیز امکان‌پذیر باشد. در این مورد در فصل‌های آینده صحبت خواهیم کرد.

در پایان، API‌های مرتبط با HTML5 تکنیک‌هایی را که طراحان وب سال‌هاست از آنها استفاده می‌کنند را ارتقا می‌دهد. بسیاری از امور اکنون ساده‌تر شده و قدرت بیشتری در اختیار طراح قرار گرفته است. علاوه بر آن، معرفی قابلیت پیوست فیلم و موسیقی در HTML5 بدین معنی است که وابستگی به نرم‌افزارهای جانبی^۲ و افزونه‌ها^۳ که امکان پخش فیلم و موسیقی را در وب فراهم می‌کنند، کمتر خواهد شد.

در مجموع، دلایل قانع‌کننده‌ای وجود دارد که نگاهی به قابلیت‌های HTML5 و API‌های مربوط به آن بیاندازید. دلایل بیشتری نیز در طول کتاب آورده می‌شود.

-
1. semantic
 2. Third party software
 3. plugins

CSS3 چیست؟

بخش مجزای دیگر - که البته اهمیت بسیاری دارد - که در طراحی صفحات وب از آن استفاده می‌شود، برگه‌های سبک سلسله مراتبی^۱ است. همانطور که به احتمال زیاد مطلع هستید، CSS زبانی است که با استفاده از آن می‌توانیم سبک و جلوه عناصر HTML را در صفحه خود مشخص کنیم. CSS3 آخرین نسخه از مستندات CSS است. عبارت CSS3 تنها به ویژگی‌های جدید این زبان اطلاق نمی‌شود بلکه این بدان معنی است که این نسخه سومین مرحله از توسعه مستندات CSS است.^۲



نکته از مترجم

برای CSS اغلب معادل فارسی برگه سبک آبخاری و یا سبک‌نامه‌ی آبخاری در نظر گرفته شده است. من از عبارت برگه‌ی سبک سلسله مراتبی استفاده کرده‌ام. منظور از واژه cascade این است که دستورات در CSS به سبکی عمل می‌کنند که تحت شرایط مختلف، دستورات می‌توانند بر روی هم تأثیر بگذارند و دستور قوی‌تر می‌تواند اثر دستور ضعیف‌تر را خنثی کند. کلمه‌ی آبخار به درستی این مفهوم را منتقل نمی‌کند.

CSS3 حاوی تمامی آن دستوراتی است که در نسخه CSS2.1 وجود دارد. (نسخه 2.1 نسخه قدیمی مستندات CSS است). همچنین در این نسخه امکاناتی اضافه شده است که برخی از مشکلات طراحان را برطرف می‌کند. به عنوان مثال برخی مواقع طراحان مجبور بودند برای طراحی بخشی خاص، عناصر HTML اضافه‌ای را در صفحه خود قرار دهند تا به جلوه خاص خود برسند و یا برای رسیدن به جلوه‌ای خاص مجبور بودند کدنویسی‌های پیچیده‌ای انجام دهند. گاهی اوقات نیز باید برای دستیابی به طرحی خاص، تصاویر متعددی را در صفحه به کار می‌گرفتند.

قابلیت‌های CSS3 عبارتند از پشتیبانی از گزینش‌گرهای آ‌های بیشتر، سایه انداختن زیر متن و عناصر، گوشه‌های گرد، تصاویر پس‌زمینه متعدد، انیمیشن، شفافیت و بسیاری از قابلیت‌های دیگر.

CSS3 از HTML5 مجزاست. در این کتاب، زمانی که از عبارت CSS3 استفاده می‌کنیم، منظورمان نسخه سوم از مستندات CSS است و تمرکزمان نیز به طور خاص بر روی قابلیت‌های جدید CSS3 است. از این رو CSS3 از HTML5 و API‌های مرتبط با آن مجزاست.

چرا CSS3 باید برایم مهم باشد؟

در ادامه این کتاب، به طور مفصل به قابلیت‌های جدید CSS3 می‌پردازیم. در زمانی اندک به شما می‌آموزیم که چرا قابلیت‌های جدید CSS3 برای طراحان بسیار هیجان‌انگیز است.

1. Cascading Style Sheet – CSS

2. <http://www.w3.org/Style/CSS/current-work.en.html>

3. selectors

برخی از این تکنیک‌ها جای خود را در بسیاری از پروژه‌ها باز کرده‌اند. مواردی همچون سایه‌ها، توالی رنگ‌ها^۱ و گوشه‌های گرد مثال‌های خوبی به حساب می‌آیند. این موارد را در طراحی‌های جدید به دفعات، دیده‌ایم. زمانی که از این تکنیک‌ها به خوبی استفاده شود و با طراحی کلی سایت و هدف آن هماهنگی داشته باشد، می‌توانند یک طرح را از باقی طرح‌ها متمایز کنند.

شاید پیش خود بگویید همین جلوه‌ها را سال‌هاست که با استفاده از CSS نسخه 2.1 ایجاد کرده‌اید، ولی آیا واقعاً اینگونه بوده است؟

در گذشته، برای ایجاد یک توالی رنگ، سایه و یا گوشه‌های گرد، طراحان مجبور بودند از ترفندهای خاصی استفاده کنند. گاهی اوقات برای پیاده‌سازی این جلوه‌ها لازم بود که عناصر HTML اضافه‌ای به صفحه افزوده شود. در آغاز طراحی، کد صفحه بسیار تمیز بود، ولی برای ایجاد این جلوه‌ها باید به کدنویسی زیاد و استفاده از ترفندها روی آورده می‌شد. برای ایجاد توالی رنگ، راهی به جز استفاده از تصویر وجود نداشت. ما از این ترفندها استفاده می‌کردیم، زیرا راه دیگری برای رسیدن به این جلوه‌ها وجود نداشت.

CSS3 به شما اجازه می‌دهد همین جلوه‌ها و حتی جلوه‌هایی دیگر را با نگاهی آینده‌نگرانه به صفحه خود اضافه کنید که این امر منجر به منافع بسیاری خواهد شد از جمله اینکه: کد صفحه شما تمیزتر خواهد ماند و برای انسان و ماشین در دسترس‌تر خواهد بود، نگهداری از کد آسان‌تر می‌شود، تصاویر کمتری در طرح استفاده می‌شود و زمان بارگذاری صفحه نیز کوتاه‌تر می‌شود.



خوب است بدانید: نکته در مورد پیشوند شرکت‌های سازنده مرورگر

برای اینکه بتوانید از بسیاری از قابلیت‌های CSS3 استفاده کنید، باید چند خط کد اضافه بنویسید. به این دلیل که شرکت‌های سازنده مرورگر، به دستور مربوط به هر کدام از قابلیت‌ها، پیشوندی خاص اضافه کرده‌اند. برای مثال، برای اینکه از قابلیت transform در فایرفاکس استفاده کنید، باید از دستور moz-transform استفاده کنید، در حالی که برای ایجاد همین جلوه برای مرورگرها برپایه WebKit مانند مرورگر سفاری و کروم، باید از دستور webkit-transform استفاده شود. گاهی اوقات برای استفاده از برخی از این قابلیت‌ها باید چهار خط کد بنویسید که همه این چهار خط فقط معرف یک دستور است که با چهار پیشوند مشخص شده است. شاید به نظر برسد که با این کار، برخی از منفعتهای مربوط به کدنویسی خلاصه، از بین رفته است.

اما شرکت‌های سازنده مرورگر قابلیت پیشوندها را به دلیلی کاملاً منطقی ایجاد کرده‌اند: مستندات مربوط به CSS3 هنوز کامل نشده است و اضافه کردن این قابلیت‌ها ممکن است همراه با اشکال باشد. از این رو در حال حاضر می‌توانید برای استفاده از قابلیت‌های خاص از دستور مربوط به آن به همراه پیشوند مربوط به مرورگر مربوطه استفاده کنید و همچنین در آخر هم همین دستور را بدون پیشوند بنویسید. زمانی که مستندات کامل شوند و پیاده‌سازی این قابلیت‌ها در مرورگرها کامل شود، این پیشوندها توسط شرکت‌های سازنده مرورگر حذف خواهند

1. gradient

شد.

با این حال به نظر می‌رسد که نگهداری کدهای CSS به همراه همه این پیشوندها کار سختی باشد، ولی باز با این وجود، در حال حاضر مزایای استفاده از CSS3 از معایبش بیشتر است. برای اینکه بخواهید یک جلوه خاص را تغییر دهید شاید مجبور باشید کدهای مربوط به یک دستور خاص که چندین بار تکرار شده و پیشوندهای مختلف دارد را تغییر دهید، ولی با این حال نگهداری طرحی که کاملاً بر پایه CSS است به مراتب آسان‌تر از این است که بخواهید همین جلوه‌ها را با استفاده از ایجاد تغییر در تصاویر گرافیکی، یا با استفاده از کدهای اضافه و کدنویسی‌های پیچیده انجام دهید. علاوه بر این، همانطور که توضیح دادیم، با این شیوه کدنویسی، کد شما تمیزتر بوده و به خاطر استفاده از استانداردهای جدیدتر، دیرتر قدیمی شده و از رده خارج می‌شود.

منظور ما از یادگیری در قالب پروژه چیست؟

در دنیای واقعی، ما سایت را بعد از طراحی رها نمی‌کنیم تا پروژه‌های جدیدتر را شروع کنیم، بلکه ما برنامه‌های تحت وبی را که خلق می‌کنیم، به‌طور مداوم به روزشان می‌کنیم، ویرایش‌شان می‌کنیم تا کارآمدتر شوند، تست‌شان می‌کنیم تا ایرادات بالقوه‌ی مربوط به کارآیی سیستم را بیابیم و به‌طور پیوسته بر روی طراحی، شکل و شمایل و چینش اجزای صفحه و حتی محتوای سایت کار می‌کنیم و آن را بهبود می‌بخشیم.

به زبان دیگر، در دنیای واقعی، کدی را نمی‌نویسیم که به حال خودش رهاش کنیم. ما برای کدنویسی از قابل اعتمادترین و مؤثرترین شیوه‌های کدنویسی استفاده می‌کنیم و هدفمان نیز این است که دوباره به کدمان رجوع کنیم و آن را بهتر کرده و تغییرش دهیم. این رویه را نه تنها برای وب‌سایت‌ها و برنامه‌های تحت وب استفاده می‌کنیم، بلکه در زمان طراحی سایت برای مشتریانمان نیز این اصول را رعایت می‌کنیم.

ما احتیاج داریم که به شکل مستمر به دنبال شیوه‌های بهتر برای نوشتن کدهایمان باشیم. استفاده از HTML5 و CSS3 گامی بزرگ در جهت نوشتن کدی قابل اعتماد، مرتب و مناسب است. ما در این کتاب سعی داریم که در قالب یک پروژه و درست همانند شرایط واقعی، استفاده از این استانداردها را به شما بیاموزیم.

بازار مرورگرهای مختلف

اگرچه HTML5 هنوز در حال توسعه است و استفاده از آن باعث تغییر عظیم در شیوه کدنویسی می‌شود، اما لازم است خاطر نشان کنیم که این تغییرات در شیوه کدنویسی باعث ایجاد مشکل در شیوه نمایش، ایجاد خطا و یا غیرقابل استفاده شدن طرح در مرورگرهای قدیمی نمی‌شود.

این بدان معنی است که شما می‌توانید یکی از پروژه‌های اخیرتان را که با HTML4 و یا XHTML نوشته‌ای را باز کنید و نوع سند¹ را به HTML5 تغییر دهید. (در این مورد در فصل 2 بحث خواهیم کرد)، پس از این کار صفحه شما همچنان معتبر و سالم است و درست همانند قبل در مرورگر نمایش داده خواهد شد. تغییرات و اضافات HTML5 به شکلی تعبیه شده‌اند که به شما اطمینان می‌دهند، طرح شما با مرورگرهای قدیمی همچون IE6 هم سازگار است.

تاکنون فقط در مورد کدنویسی صحبت کردیم. وضعیت دیگر قابلیت‌های HTML5 و CSS3 و دیگر فناوری‌های مرتبط با آن چگونه است؟ براساس آمار²، حدود 47 درصد از کاربران هنوز از یکی از نسخه‌های اینترنت اکسپلورر استفاده می‌کنند که از بیشتر قابلیت‌های جدید هیچ پشتیبانی‌ای نمی‌کند.



نکته از مترجم

البته سهم مرورگر اینترنت اکسپلورر در حال کاهش است و در زمان ترجمه این کتاب و تا آغاز ماه دسامبر سال 2011، سهم این مرورگر به حدود 40 درصد رسیده است. مرورگر کروم به سرعت در حال محبوب شدن است و اکنون بعد از اینترنت اکسپلورر در جایگاه دوم قرار دارد و با اختلافی اندک، مرورگر فایرفاکس را نیز پشت سر گذاشته است. وبلاگ انتشارات سایت پوینت، اطلاعات مربوط به سهم مرورگرها را در آغاز هر ماه، منتشر می‌کند. برای کسب اطلاعات در این زمینه می‌توانید به سایت انتشارات سایت پوینت در آدرس <http://www.sitepoint.com/> مراجعه کنید.

در نتیجه، برنامه‌نویسان، راهکارهای جایگزینی را ابداع کرده‌اند تا از تمامی قابلیت‌های HTML5 و CSS3 استفاده کنند و در عین حال تجربه‌ای مشابه را در اختیار کاربر قرار دهند. گاهی اوقات این فرآیند ساده است. مانند زمانی که شما نخست کنترل می‌کنید که مرورگر کاربر قابلیت پشتیبانی از پیوست-کردن فیلم با استفاده از برچسب‌های جدید HTML5 را دارد یا خیر؛ در صورتی که مرورگر از این قابلیت پشتیبانی نمی‌کند، نسخه Flash فیلم موردنظر خود را در اختیار مرورگر قرار می‌دهید. گاهی اوقات فرآیند کمی پیچیده‌تر است و باید با استفاده از اسکریپت‌نویسی، همان جلوه موردنظر را در مرورگری که از قابلیت موردنظر شما پشتیبانی نمی‌کند، شبیه‌سازی کنید. به تکنیک‌های شبیه‌سازی قابلیت‌های جدید، pollyfill می‌گویند. متکی بودن به شبیه‌سازی قابلیت‌های جدید با استفاده از اسکریپت‌نویسی، همیشه بهترین راه محسوب نمی‌شود، مخصوصاً زمانی که قصد توسعه یک ابزار تحت وب را دارید که در آن، کارایی و سرعت اهمیت بسیار دارد. اما گاهی اوقات تحمل زجر ناشی از اضافه کردن این قابلیت‌ها ضروری است و ما در این کتاب مثال‌های متعددی را در این مورد، ذکر می‌کنیم. از این رو، زمانی که قصد داریم تکنیک‌های جایگزین و pollyfillها را به شما معرفی کنیم، نهایت سعی‌مان را می‌کنیم تا شما را از کاستی‌ها و نقاط ضعف هر کدام از این تکنیک‌ها آشنا کنیم.

1. Doctype

2. http://gs.statcounter.com/#browser_version-ww-monthly-201011-201101-bar

قطعاً این نکته را باید اطلاع دهیم که در بسیاری از مواقع به تکنیک‌های جایگزین و `polyfill`ها احتیاجی نداریم. به عنوان مثال، زمانی که برای ایجاد گوشه‌های گرد از CSS3 استفاده کنیم، مرورگرهای قدیمی، آن عناصر موردنظر ما را با گوشه‌های تیز نمایش می‌دهند، ولی این تفاوت نمایش در مرورگرهای جدید و قدیمی، اینقدر اهمیت ندارد که برای ایجاد آنها در مرورگرهای قدیمی بخواهیم از تکنیک‌های جایگزین استفاده کنیم. با اینکه گوشه‌های عناصر در مرورگرهای قدیمی همچنان تیز نمایش داده می‌شود، ولی کل سایت قابل استفاده است و در نهایت، استفاده از سایت با خلل مواجه نشده است. حتی ممکن است که کاربر متوجه تفاوت طرح در مرورگرهای قدیمی و جدید نشود.

با دانستن این موضوع که پشتیبانی مرورگرها از این استانداردهای جدید، محدود است، ممکن است انگیزه خود را از دست داده باشید. اینطور فکر نکنید! خبر خوب این است که بیش از 40 درصد کاربران در سراسر دنیا از مرورگرهای جدید استفاده می‌کنند که از بیشتر قابلیت‌هایی که در این کتاب در موردشان صحبت می‌شود، پشتیبانی می‌کنند. این پشتیبانی از قابلیت‌های جدید نیز دائماً در حال افزایش است و با ارائه نسخه جدید هر مرورگر، قابلیت‌های بیشتری توسط آن مرورگر پشتیبانی می‌شود. (مانند نسخه 9 از مرورگر اینترنت اکسپلورر) علاوه بر آن، این چرخه نیز ادامه‌دار است و متوقف نمی‌شود.

همچنان که در کتاب به پیش می‌رویم، به شما اطلاع خواهیم داد که پشتیبانی از کدام یک از قابلیت‌ها کامل نیست، از این رو مطلع می‌شوید که چه مقدار از طرحی که با استفاده از HTML5 و CSS3 طراحی کرده‌اید، توسط کاربران، قابل دیدن و استفاده کردن است. همچنین شیوه‌های جایگزین را هم به شما معرفی می‌کنیم که بتوانید با استفاده از آنها، تجربه‌ای قابل قبول را نیز در اختیار کاربرانی قرار دهید که از مرورگرهای قدیمی استفاده می‌کنند و نمی‌توانند از قابلیت‌های HTML5 و CSS3 بهره ببرند.

رشد سریع وب‌گردی از طریق تلفن همراه

یکی از دلایل قانع‌کننده‌ی دیگر برای استفاده از HTML5 و CSS3، رشد سریع وب‌گردی از طریق تلفن همراه است.

براساس آماری که توسط سایت StatCounter در سال 2009 اعلام شده است، اندکی بیش از یک درصد از کاربران وب از تلفن‌همراه خود برای گشت و گذار در وب، استفاده می‌کنند¹. در کمتر از دو سال، این مقدار چهار برابر شده است و از چهار درصد گذشته است². برخی مدعی شده‌اند که آمار واقعی بیش از این مقدار است و اگر از تکنیک‌های دیگری برای تحلیل آمار استفاده شود، مقدار متفاوت خواهد بود. حال هرچه باشد، واضح است که بازار تلفن‌های همراه و وب‌گردی از طریق گوشی موبایل، به طرز شگفت‌انگیزی در حال رشد است.

1. http://gs.statcounter.com/#mobile_vs_desktop-ww-monthly-200901-200912-bar

2. http://gs.statcounter.com/#mobile_vs_desktop-ww-monthly-201011-201101-bar

شاید مقدار 4 درصد از نظر شما مقدار بسیار کمی به حساب بیاید و قطعاً نیز همین‌طور است. اما رشد این مقدار بسیار قابل توجه است؛ 400 درصد رشد در عرض دو سال! خوب حال این چه ارتباطی با یادگیری HTML5 و CSS3 دارد؟

HTML5, CSS3 و فناوری‌های مرتبط با آن به شکل بسیار خوبی در مرورگرهای موجود بر روی گوشی‌های موبایل، پشتیبانی شده‌اند. برای مثال، مرورگر سفاری مخصوص گوشی موبایل بر روی ابزارهایی که از سیستم عامل iOS استفاده می‌کنند. (مانند ابزارهای iPhone و iPad) مرورگر اپرا مینی و اپرا موبایل و همین‌طور مرورگر موجود بر روی سیستم عامل اندروید¹ به خوبی از استانداردهای HTML5 و CSS3 پشتیبانی می‌کنند. قابلیت‌های جدید که توسط برخی از این مرورگرهای پشتیبانی می‌شوند عبارتند از: رنگ‌ها و قابلیت شفافیت در CSS3، پشتیبانی از Canvas API، Web Storage، SVG، گوشه‌های گرد در CSS3، ذخیره‌سازی Offline داده‌ها، ابزارهای Offline² و بسیاری از قابلیت‌های جدید دیگر.

در حقیقت، بسیاری از فناوری‌هایی که ما در این کتاب به شما معرفی می‌کنیم، به طور خاص برای استفاده در گوشی‌های موبایل طراحی شده‌اند. فناوری‌هایی مانند ابزارهای Offline و Web Storage به این خاطر طراحی شده‌اند تا نیاز کاربرانی که از طریق گوشی موبایل به اینترنت مراجعه می‌کنند را برآورده کنند. گوشی‌های موبایل اغلب دسترسی محدودی به اینترنت دارند و با استفاده از این فناوری‌ها، می‌توان در زمانی که دسترسی به اینترنت وجود نداشت، از این ابزارها استفاده کرد.

در فصل 10 در مورد این فناوری‌ها بحث خواهیم کرد؛ در طول کتاب نیز با فناوری‌های دیگری آشنا می‌شوید که با استفاده از آنها می‌توانید برنامه‌هایی را خلق کنید که در ابزارهای مختلفی بتوان از آنها استفاده کرد.

طراحی برای شرایط واقعی

معقول نیست که بخواهیم از یک فناوری جدید استفاده کنیم و فقط بخواهیم که سایتمان در یک سری مرورگر خاص قابل استفاده باشد. در دنیای واقعی، و در شرایطی که بخواهیم HTML5 و CSS3 نفوذ بیشتری در بین طرح‌ها پیدا کند، باید صفحات سایتمان را به شکلی طراحی کنیم که در محیط‌های مختلف، قابل استفاده باشد. این محیط می‌تواند مرورگرهای جدید، مرورگرهای قدیمی و یا حتی مرورگرهای موجود بر روی گوشی تلفن همراه باشد.

درست است، شناسایی مرورگرهای قدیمی و شبیه‌سازی امکانات جدید در آنها شما را یاد نخستین روزهای وب می‌اندازد که استانداردهای وب هنوز تدوین نشده بود و مجبور بودید از طریق زبان‌های برنامه‌نویسی، مرورگرها را شناسایی کنید و نسخه‌های مختلفی از سایت را به هر کدام از آنها نشان دهید. اما به یاد داشته باشید که در زمان حال، کدی که می‌نویسید براساس استانداردهایی است

1. Android

2. Offline apps

که در آینده نزدیک جایگزین استانداردهای موجود خواهند شد، از این رو زمانی که مرورگرهای قدیمی، از رده خارج شوند، تنها کاری که باید انجام دهید این است که تکنیک‌های جایگزین و polyfill را حذف کنید و تنها کدهایی را باقی بگذارید که برای مرورگرهای جدید نوشته شده بودند. HTML5 و CSS3 سردمدار فناوری‌های جدید وب‌اند که دنیای جذاب و جدیدی را برای طراحان وب ایجاد کرده‌اند. از آنجاییکه مرورگرهای پیشرفته (از جمله اینترنت اکسپلورر 9) از برخی از قابلیت‌های HTML5 و CSS3 پشتیبانی خوبی به عمل می‌آورند، ساخت یک صفحه وب قدرتمند، با قابلیت نگهداری آسان و همینطور سازگاری بالا با استانداردهای آینده، در حال حاضر از هر زمان دیگری برای طراحان وب آسان‌تر است.

مهارتی که امروز در استفاده از HTML5 و CSS3 کسب می‌کنید در زمانی که میزان استفاده از مرورگرهای قدیمی، کاهش یابد، برایتان بسیار ارزشمند می‌شود. امروز با یادگیری این فناوری‌ها خود را برای آینده‌ای درخشان در زمینه‌ی طراحی وب، آماده می‌کنید. خوب دیگر بیان دلایل منطقی کافیست، حال اجازه بدهید برایتان توضیح دهید که چگونه می‌توان از این استانداردها استفاده کرد!

فصل هفتم

توالی رنگ^۱ با CSS3 و تصاویر پس‌زمینه‌ی

متعدد

در فصل ششم روش‌هایی برای اضافه‌کردن سبک‌های زیبا مثل اضافه‌کردن سایه و گوشه‌های گرد به عناصر را یاد گرفتیم بدون اینکه نیازی باشد، کدی را به صفحه اضافه کنیم و یا تصاویر را به کار ببریم. جلوه‌ی دیگری که بسیار در طراحی رایج است و برای دستیابی به آن به تصاویر احتیاج داریم، توالی رنگ‌ها هستند. CSS3 قابلیت‌ی را در اختیار ما قرار می‌دهد که توسط آن توالی رنگ‌های خطی و دایره‌ای بسازیم و همچنین چندین تصویر را به طور همزمان در پس‌زمینه یک عنصر قرار دهیم. با CSS3 دیگر لازم نیست که چندین توالی رنگ مختلف را با فرمت JPEG بسازیم و یا برای قراردادن چند تصویر در پس‌زمینه، کدهایی بی‌ارزش را به صفحه خود اضافه کنیم.

پشتیبانی مرورگرها از توالی رنگ‌ها و اضافه کردن چند تصویر پس‌زمینه در حال تحول است، اما در ادامه این فصل خواهید دید که می‌توان از توالی رنگ‌ها استفاده کرد. به طوری که آخرین نسخه تمامی مرورگرهای مطرح و همچنین IE9 بتوانند آنها را به نمایش بگذارند.

با توالی رنگ‌ها در CSS3 شروع می‌کنیم، اما نخست اجازه دهید که ببینیم، توالی رنگ چیست؟ **توالی رنگ**، شامل طیفی پیوسته از رنگ و تغییر آن از یک رنگ به رنگ و یا رنگ‌های دیگر است. برای ساخت توالی رنگ، شما می‌توانید چندین رنگ را مشخص کنید که به آنها، توقف رنگ^۲ می‌گویند. هر کدام از این توقف رنگ‌ها شامل یک رنگ و یک موقعیت می‌شود؛ مرورگر، رنگ را از هر توقف رنگ به سمت توقف رنگ بعدی، محو می‌کند تا یک توالی رنگ نرم را بسازد. در هر جایی که توانایی استفاده از یک تصویر پس‌زمینه را دارید، می‌توانید از توالی رنگ نیز استفاده کنید. این بدان معناست که هر کجا در CSS که از دستور url() استفاده می‌کردید (مانند دستورات background-image, border-image و حتی list-

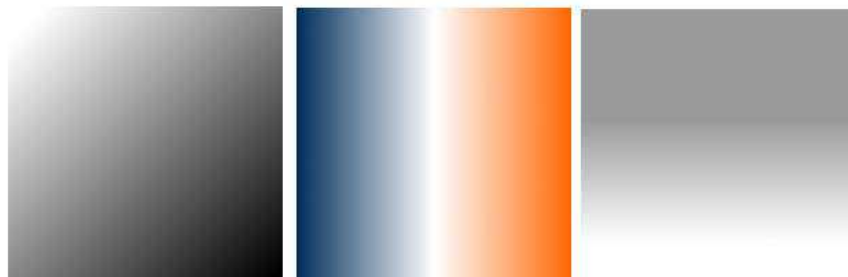
1. gradient
2. Color stop

style-type) می‌توانید از توالی رنگ‌ها نیز استفاده کنید. البته در حال حاضر، استفاده از توالی رنگ‌ها در background-image بهترین پشتیبانی را دارد.

با استفاده از توالی رنگ‌ها در CSS و حذف تصاویر، کاربران دیگر مجبور نخواهند بود که برای دیدن جلوه توالی رنگ، تصاویر توالی رنگ را دریافت کنند و همچنین انعطاف طرح شما نیز بیشتر می‌شود و با بزرگنمایی طرح، توالی رنگ‌های CSS دیگر همانند تصاویر پس‌زمینه، بی‌کیفیت نمی‌شوند. دو نوع توالی رنگ در CSS3 در حال حاضر موجود است: خطی و دایره‌ای. اجازه دهید آنها را به ترتیب برایتان شرح دهیم.

توالی رنگ خطی

توالی رنگ خطی آن توالی رنگ‌هایی است که رنگ‌ها در طول یک خط، از رنگی به رنگ دیگر، تغییر می‌کنند: به عنوان مثال از بالا به پایین، از چپ به راست و یا در قطر چهار ضلعی. اگر با برنامه‌های گرافیکی مانند فوتوشاپ یا فایر ورکز کار کرده باشید، به احتمال زیاد با توالی رنگ خطی، آشنایی دارید. در تصویر 1-7 مثال‌هایی از این نوع توالی رنگ، نمایش داده شده است.



تصویر 1-7 مثال‌هایی از توالی رنگ خطی

همانند برنامه‌های ویرایش تصویر، برای ساخت یک توالی رنگ باید جهت، رنگ آغازین و رنگ پایانی، و توقف رنگ‌هایی را که مایلید در طول این توالی رنگ، ایجاد شود را مشخص کنید. مرورگر، باقی کارها را برای شما انجام می‌دهد و تمامی سطح عنصر موردنظر شما را با توالی رنگ، می‌پوشاند. این توالی، از یک رنگ شروع شده و به نرمی به رنگ بعد می‌رسد و درست در جهتی حرکت می‌کند که شما مشخص کرده‌اید.

زمانی که بحث توالی رنگ خطی در مرورگر مطرح می‌شود، شرایط کمی به هم ریخته به نظر می‌رسد. WebKit اولین بار قابلیت ترسیم توالی رنگ را، سال‌ها پیش معرفی کرد و از ساختاری برای تعریف دستور استفاده کرد که به گفته بسیاری، گنگ و پیچیده است. پس از آن موزیلا قابلیت توالی رنگ را در مرورگر خود پشتیبانی کرد و از ساختاری ساده و مختصرتر استفاده کرد. سپس در ژانویه 2011 کنسرسیوم جهانی وب (W3C) ساختار پیشنهادی توالی رنگ در CSS3 را به مستندات اضافه کرد. ساختار تازه با ساختاری که در فایرفاکس موجود بود، شباهت بسیاری داشت. در حقیقت این شباهت

به قدری زیاد است که اگر دستورات توالی رنگ را با استاندارد W3C بنویسید، توالی رنگتان بدون مشکل در فایرفاکس نمایش داده می‌شود. ساختار جدید W3C در نسخه‌های جدید مرورگرهای WebKit پشتیبانی شده، ولی در زمان نگارش این کتاب، تنها در نسخه تحت توسعه، این قابلیت اضافه شده است و تا زمان به کارگیری آنها در مرورگرهای کروم و سفاری، اندکی زمان لازم است و در حال حاضر این مرورگرها از دستورات قدیمی توالی رنگ، پشتیبانی می‌کنند. برای پشتیبانی از سایت‌هایی که با دستورات قدیمی توالی رنگ، طراحی شده‌اند، این مرورگرها تا مدتی، همچنان دستورات قدیمی را پشتیبانی خواهند کرد. مرورگر آپرا نیز از نسخه‌ی 11.10 این ویژگی را براساس استاندارد W3C پشتیبانی می‌کند. در حال حاضر تمامی مرورگرهای فوق، از این دستورات با پیشوند مخصوص خودشان پشتیبانی می‌کنند. (-webkit-, -moz-, -o-)



خوب است بدانید – WebKit نسخه‌ی روزانه^۱

موتور پردازش WebKit که در قلب مرورگرهای سفاری و کروم قرار گرفته است، یک پروژه مستقل و منبع باز^۲ است و سایت این پروژه نیز از آدرس <http://www.webkit.org/> در دسترس است. اگرچه قابلیت‌های جدید به سرعت به WebKit اضافه می‌شود، ولی تا زمان استفاده از این موتور جدید در مرورگرهای سفاری و کروم، اندکی زمان لازم است. اما با این حال می‌توانید با نصب یکی از نسخه‌های روزانه‌ی مرورگرها، این قابلیت را تست کنید. از این جهت از واژه‌ی روزانه استفاده می‌شود، زیرا این نسخه از مرورگر به طور روزانه منتشر می‌شود و حاوی آخرین تغییراتی است که توسط برنامه‌نویسان بر روی، مرورگر اعمال شده است. از آنجایی که این نسخه از مرورگر به طور مستمر و هر روزه منتشر می‌شود و تحت توسعه است، ممکن است حاوی قابلیت‌ی نیمه‌کاره باشد و یا معایبی در عملکرد مرورگر وجود داشته باشد به همین خاطر، این نسخه‌ها، با ثبات نیستند. با این حال، اگر بخواهید قابلیت‌ی جدید را تست کنید (به عنوان مثال، شیوه‌ی نگارش توالی رنگ براساس استاندارد W3C) که هنوز در نسخه‌های موجود کروم و سفاری در دسترس نیست، می‌توانید از این نسخه‌ها استفاده کنید. برای دریافت آخرین نسخه‌ی روزانه‌ی WebKit برای سیستم‌عامل ویندوز و مکینتاش، به آدرس <http://nightly.webkit.org/> مراجعه کنید.

با این حال این سؤال باقی می‌ماند که چطور توالی رنگ‌ها را در IE و نسخه‌های قدیمی Opera به کار بگیریم. خوشبختانه IE9 و آپرا در نسخه 11.10 و نسخه‌های قدیمی‌تر از قابلیت SVG در پس‌زمینه، پشتیبانی می‌کنند و ساخت تصاویر توالی رنگ در SVG بسیار آسان است. (در مورد SVG در فصل یازدهم به طور مفصل، بحث خواهیم کرد.) و در نهایت، تمامی نسخه‌های مرورگر IE از دستور اختصاصی برای ایجاد توالی رنگ خطی، پشتیبانی می‌کنند.

1. Nightly Build
2. Open source

گیج شده‌اید؟ نگران نباشید. اگرچه درک شیوه‌ی استفاده از توالی رنگ‌ها مهم است، اما به خاطر سپاری شیوه‌ی نگارش دستورات، الزامی نیست. ما الگوی نگارش دستور جدید و همچنین الگوی قدیمی مرورگر WebKit را که به زودی فراموش خواهد شد را نیز توضیح می‌دهیم و سپس رازی را با شما در میان می‌گذاریم: یک ابزار که می‌تواند دستورات توالی رنگ را برای شما بنویسد و از این طریق دیگر نیازی نیست که دستورات مخصوص هر مرورگر را به خاطر بسپارید. اجازه دهید شروع کنیم.

تنها یک توالی رنگ خطی در سایت HTML5 herald وجود دارد و آن هم در مستطیل مربوط به دومین جعبه تبلیغاتی در سایت است که در تصویر 2-7 نیز (که حاوی تبلیغ مربوط به همین کتاب است!) نیز نمایش داده شده است. احتمالاً متوجه شده‌اید که توالی رنگ از قسمت بالا و با رنگ سیاه شروع می‌شود و به سمت روشنی پیش می‌رود و سپس دوباره تیره می‌شود و مسیر را در زیر چرخ‌های دوچرخه، رسم می‌کند و دوباره روشن می‌شود.



تصویر 2-7 یک توالی رنگ خطی در سایت HTML5 herald

برای ساخت توالی رنگ برای بخش تبلیغات، نخست با شیوه نگارش استاندارد دستور توالی رنگ، شروع می‌کنیم. این شیوه نگارش از همه ساده‌تر و قابل فهم‌تر است و تنها دستوری خواهد بود که در آینده از آن استفاده خواهید کرد. سپس شیوه نگارش دستور به سبک قدیمی در WebKit و فایرفاکس می‌پردازیم و تفاوتشان را با دستور جدید بررسی می‌کنیم.

دستور استاندارد W3C

این دستور استاندارد برای ایجاد توالی رنگ خطی است:

```
background-image: linear-gradient( ... );
```

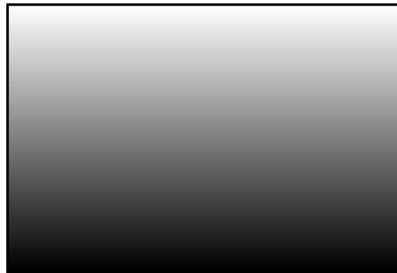
در داخل آن پرانتزها، جهت ایجاد توالی رنگ، و توقف رنگ‌ها را مشخص می‌کنید. برای تعیین جهت هم می‌توانید زاویه را که مایلید، توالی رنگ در راستای آن حرکت کند را مشخص کنید و هم می‌توانید با استفاده از کلمات کلیدی، آن گوشه‌ای را که می‌خواهید توالی رنگ از آنجا شروع شود را مشخص کنید. در هر یک از این دو حالت، توالی رنگ به سمت مخالف جهت تعیین شده توسط شما، حرکت خواهد کرد. برای تعیین زاویه، باید مقادیر را به درجه بنویسید (deg). (deg) به سمت راست اشاره می‌کند، 90deg به

سمت بالا و به همین ترتیب بقیه‌ی مقادیر در راستای جهت عقربه‌های سایت، جلو می‌روند. برای سمت‌ها و گوشه‌ها نیز از کلمات کلیدی top, bottom, left و right استفاده می‌شود. پس از مشخص کردن جهت، توقف رنگ‌ها را تعیین می‌کنید. این بخش از کد، رنگ‌ها و درصد و یا طول مربوط به گسترش آن رنگ را مشخص می‌کند.

مطالب بسیاری است که باید یاد بگیرید، به همین خاطر بهتر است که نخست چند مثال از توالی رنگ‌ها را ببینید. به‌منظور نمایش‌گذاری توالی رنگ، مثالی را با استفاده از دو رنگ #FFF (سفید) و #000 (سیاه) ایجاد می‌کنیم.

برای رسم توالی رنگ از بالا به پایین در پس‌زمینه‌ی یک عنصر (همانند تصویر 7.3) باید یکی از سه دستور زیر را به کار ببرید. (ما در مثالمان از پیشوند -moz- استفاده کرده‌ایم، ولی به یاد داشته باشید که نسخه‌های جدید مرورگرها برپایه‌ی WebKit و مرورگر اپرا هر دو از پیشوندهای خودشان یعنی -webkit- و -o- استفاده می‌کنند، ولی الگوی نگارش دستور در هر دویشان یکسان است):

```
background-image: -moz-linear-gradient(270deg, #FFF 0%, #000 100%);
background-image: -moz-linear-gradient(top, #FFF 0%, #000 100%);
background-image: -moz-linear-gradient(#FFF 0%, #000 100%);
```



تصویر 7-3 توالی رنگ سفید به سیاه که از بالا و مرکز عنصر شروع شده و به پایین و مرکز عنصر ختم شده است.

آخرین دستوری که نوشتیم به درستی کار می‌کند، زیرا مقدار top که در دستور آخر قید نشده است، مقدار پیش‌فرض است و در صورتی که این مقدار را مشخص نکنید، مرورگر به طور خودکار، مقدار top را در نظر خواهد گرفت.

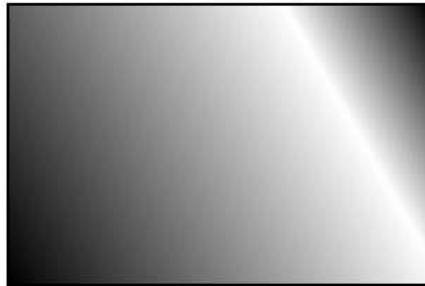
از آنجایی که اولین توقف رنگ در 0 درصد اتفاق می‌افتد و آخرین توقف رنگ نیز در 100 درصد اتفاق افتاده است، می‌توان درصدهایی که در مثال قید شده است را حذف کرد و نتیجه همین چیزی خواهد بود که می‌بینیم:

```
background-image: -moz-linear-gradient(#FFF, #000);
```

حال، اجازه دهید که توالی رنگ را بر روی یک درجه‌ی خاص، ایجاد کنیم و چند توقف رنگ نیز به آن اضافه کنیم. فرض کنید که می‌خواهیم از رنگ سیاه به سفید برویم و دوباره به رنگ سیاه بازگردیم:

```
background-image: -moz-linear-gradient(30deg, #000, #FFF 75%, #000);
```

ما توقف رنگ را در 75 درصد قرار دادیم از این رو، نوار سفید رنگ، نزدیک به پایان توالی رنگ نمایان شده است. به تصویر 4-7 دقت کنید.

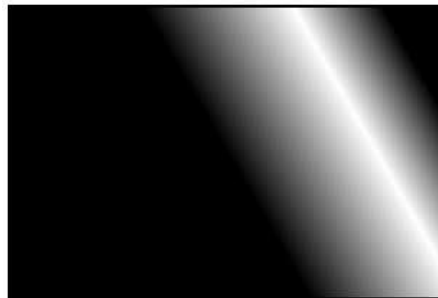


تصویر 4-7 یک توالی رنگ که دارای سه توقف رنگ است.

می‌توانید نقطه شروع اولین توقف رنگ را در نقطه‌ای به غیر از 0 درصد شروع کنید و نقطه پایان را نیز به همین شکل در جایی به غیر از 100 درصد قرار دهید. فاصله بین نقطه 0 درصد و نقطه‌ای که برای شروع انتخاب کرده‌اید، با رنگی که در توقف رنگ اول مشخص کرده‌اید، پر می‌شود و همینطور رنگی که بین نقطه پایان و مقدار 100 درصد را پر می‌کند نیز با رنگی که برای توقف رنگ دوم در نظر گرفته‌اید یکسان است. حال به این مثال توجه کنید:

```
background-image: -moz-linear-gradient(30deg, #000 50%, #FFF 75%, #000 90%);
```

توالی رنگی که در نتیجه‌ی این دستور ایجاد می‌شود، شبیه به تصویر 5-7 خواهد بود.



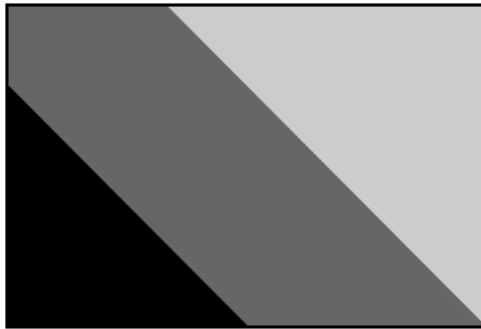
تصویر 5-7 اگر نقطه‌ی شروع و پایان را با فاصله از مقادیر 0 و 100 درصد مشخص کنید، توالی رنگی فشرده ایجاد می‌شود.

هر زمانی که یک توقف رنگ جدید تعریف می‌کنید لازم نیست که موقعیت آن را با استفاده از درصد مشخص کنید. اگر مقدار درصدی را ننویسید، فواصل به شکل خودکار بین توالی رنگ‌ها تقسیم می‌شود:

```
background-image:
-moz-linear-gradient(45deg,
```

```
#FF0000 0%,
#FF6633 20%,
#FFFF00 40%,
#00FF00 60%,
#0000FF 80%,
#AA00AA 100%);
background-image:
-moz-linear-gradient(45deg,
#FF0000,
#FF6633,
#FFFF00,
#00FF00,
#0000FF,
#AA00AA);
```

هر کدام از دو تکه کد بالا، رنگین‌کمانی را ایجاد می‌کند که اندکی زاویه دارد و ظاهرش نیز آنچنان جذاب نیست. ما دستورات را در چند خط نوشتیم و فرورفتگی‌هایی را هم در کد ایجاد کرده‌ایم تا خوانایی کدمان بیشتر شود. (اضافه کردن این فاصله‌ها و فرورفتگی‌ها الزامی نیست.) در توالی رنگ، رنگ‌ها به نرمی از یک رنگ به رنگی دیگر تغییر می‌کنند. اما اگر دو رنگ را درست در موقعیتی یکسان با هم، تعریف کنید، رنگ‌ها در همدیگر محو نخواهند شد و مرزی مشخص و واضح بین دو رنگ، شکل می‌گیرد. با استفاده از این ترفند می‌توان پس‌زمینه‌هایی راه‌راه را ایجاد کرد، درست همانند چیزی که در تصویر 6-7 می‌بینید.



تصویر 6-7 با چیش دقیق رنگ‌ها می‌توانید پس‌زمینه‌ای راه‌راه ایجاد کنید.

حال به کدی که برای ساخت تصویر قبل از آن استفاده شده، نگاهی بیندازیم:

```
background-image:
-moz-linear-gradient(45deg,
#000000 30%,
#666666 30%,
#666666 60%,
```

```
#CCCCCC 60%,
#CCCCCC 90%);
```

در آینده نزدیک، می‌توانید انتظار داشته باشید که این نسخه به روز شده از شیوه نگارش دستور، تنها شیوه‌ی معتبر و مورد نیاز خواهد بود، اما در حال حاضر به آن زمان نرسیده‌ایم و باید دستورات قدیمی را هم در کد خود بنویسیم.

دستور قدیمی در مرورگرهای برپایه‌ی WebKit

همانطور که توضیح دادیم، آخرین نسخه‌ی تحت توسعه‌ی موتور پردازش WebKit از الگوی استاندارد W3C برای نوشتن دستور توالی رنگ پشتیبانی می‌کند، با این حال نسخه‌های حال حاضر مرورگرهای برپایه‌ی WebKit، هنوز از نسخه‌ی قدیمی دستور توالی رنگ، پشتیبانی می‌کنند که کمی پیچیده‌تر است. از آنجایی‌که ما باید از مرورگرهای قدیمی نیز پشتیبانی به عمل آوریم، کدهای قدیمی را نیز در میان کدهایمان قرار می‌دهیم. اجازه دهید نگاهی سریع، به الگوی نگارش قدیمی دستور توالی رنگ، بیندازیم و مثال اولمان (توالی رنگ سفید به سیاه) را با این الگو بنویسیم:

```
background-image:
  -webkit-gradient(linear, 0% 0%, 0% 100%, from(#FFFFFF), to(#000000));
```

به جای استفاده از خصوصیت `linear-gradient`، دستوری عمومی به نام `-webkit-gradient` وجود دارد که در آن، نوع توالی رنگ را به عنوان اولین پارامتر، مشخص می‌کنید. سپس در این شیوه نگارش، باید نقطه شروع و پایان را مشخص کنید تا جهت توالی رنگ، تعیین شود. نقطه شروع و پایان را می‌توان با درصد، مقادیر عددی و یا کلمات کلیدی `top`، `bottom`، `left`، `right` و یا `center`، مشخص کرد. مرحله بعدی، مشخص کردن توقف رنگ‌های توالی رنگ است. شما می‌توانید رنگ آغازین را با کلمه کلیدی `from` و نقطه پایان را با کلمه کلیدی `to` مشخص کنید. سپس می‌توانید با استفاده از تابع `color-stop()` در صورت نیاز، توقف رنگ‌های موردنیازتان را اضافه کنید. اولین مقداری که در تابع `color-stop()` وارد می‌شود، مکان توقف رنگ را مشخص می‌کند و این مقدار نیز باید به شکل درصدی باشد و دومین مقدار نیز، رنگ را در آن نقطه، مشخص می‌کند. به این مثال دقت کنید:

```
background-image:
  -webkit-gradient(linear, left top, right bottom,
    from(red),
    to(purple),
    color-stop(20%, orange),
    color-stop(40%, yellow),
    color-stop(60%, green),
    color-stop(80%, blue));
```

با استفاده از دستور فوق، ما همان رنگین‌کمان زاویه دارمان که یادآور سایت `GeoCities` در سال 1996 است را با الگوی قدیمی نوشته‌ایم.

در حقیقت لازم نیست که نقطه شروع و پایان را با استفاده از کلمات کلیدی `from` و `to` مشخص کنیم. از آنجایی که `from(red)` با دستور `color-stop(0, red)` یکسان است، می‌توانیم دستورمان را به این شکل بنویسیم:

```
background-image:
  -webkit-gradient(linear, left top, right bottom,
    color-stop(0, red),
    color-stop(20%, orange),
    color-stop(40%, yellow),
    color-stop(60%, green),
    color-stop(80%, blue),
    color-stop(100%, purple));
```

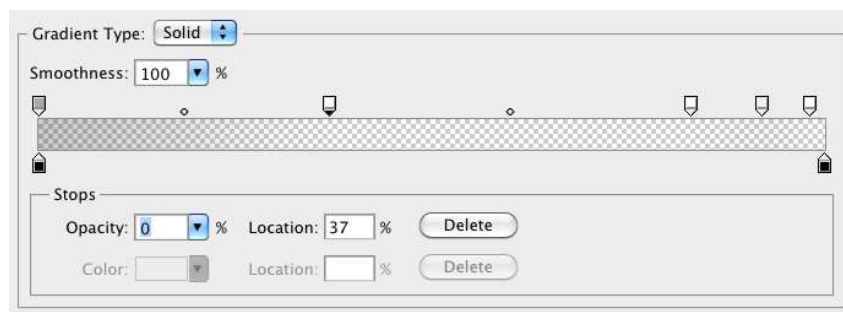
اگر از کلمه کلیدی `from` و یا یک توقف رنگ در نقطه 0 درصد استفاده نکنید، از همان رنگ مشخص شده برای اولین توقف رنگ برای پر کردن فاصله بین نقطه 0 درصد تا اولین توقف رنگ، استفاده می‌شود. در این شرایط، رنگی که حد فاصل لبه‌ی عنصر تا اولین توقف رنگ را پر می‌کند، محوشدگی ندارد و رنگی یکدست است و درست از نقطه‌ای که توقف رنگ مشخص شده است، محوشدگی به سمت رنگ دیگر، شروع می‌شود. زمانی که به آخرین توقف رنگ برسیم، در صورتی که توقف رنگ آخر در مکان 100 درصد مشخص نشده باشد، در فاصله باقیمانده‌ی بین نقطه پایانی و لبه کناری عناصر، با همان رنگ تعیین شده برای توقف رنگ پایانی، پر می‌شود. به عبارت دیگر، اگر اولین توقف رنگ را در نقطه 40 درصد مشخص کرده باشید و آخرین توقف رنگ نیز در مکان 60 درصد مشخص شده باشد، رنگ مشخص شده در اولین توقف رنگ، فاصله بین 0 تا 40 درصد را پر می‌کند و رنگ تعیین شده در توقف رنگ پایانی نیز فاصله بین 60 درصد تا 100 درصد را پر می‌کند و فاصله بین 40 در 60 درصد نیز، توالی رنگ ما و محوشدگی بین دو رنگ مشخص شده، نمایش داده می‌شود. همانطور که می‌بینید، این الگوی قدیمی از الگوی مرورگر موزیلا پیچیده‌تر است. خوشبختانه، ابزارهایی در دسترس‌اند که تمامی کدهای مربوط به توالی رنگ‌ها را برایمان تولید می‌کنند. برخی از این ابزارها را در پایان این فصل معرفی می‌کنیم. اما ببینیم که چطور این دو الگوی نگارش دستور را در سایت HTML5 herald به کار بگیریم تا توالی رنگمان در تمامی مرورگرها به درستی نمایش یابد. خبر خوب این است که مرورگرهای قدیمی برپایه WebKit برای توالی رنگ‌ها از دستور متفاوتی استفاده می‌کنند (دستور `-webkit-gradient` به جای `-webkit-linear-gradient`) می‌توانید هر دو دستور را به طور همزمان استفاده کنید، بدون اینکه تداخلی بین آنها به وجود بیاید. در حقیقت مرورگرهای برپایه‌ی WebKit جدید از هر دو الگو پشتیبانی می‌کنند، ولی از آن دستوری استفاده می‌کنند که آخر از همه آمده باشد.

کنار هم قرار دادن همه کدها

اکنون درک خوبی از شیوه‌ی نگارش دستوری توالی رنگ خطی داریم، از این رو اجازه دهید که دستورمان را بنویسیم.

اگر طراح شما در طراحی، توالی رنگ به کار برده باشد، به احتمال زیاد طراحی خود را در نرم‌افزار فوتوشاپ یا یک برنامه ویرایش تصویر دیگر انجام داده است. شما می‌توانید از این موقعیت به نفع خودتان استفاده ببرید. اگر فایل اصل مربوط به طراحی را در اختیار داشته باشید، شبیه‌سازی همان جلوه توالی رنگ داخل فایل گرافیکی با استفاده از CSS، آسان خواهد بود.

اگر فایل‌تان را در فوتوشاپ باز کنید، می‌توانید پنجره مربوط به جلوه‌ی توالی رنگ را باز کنید و کدهای رنگ مربوط به توالی رنگ بخش تبلیغات را ببینید (تصویر 7-7). توالی رنگ ما خطی است و پنج توقف رنگ نیز در آن گنجانده شده که تنها میزان شفافیت را برای رنگ سیاه، تغییر می‌دهند.



تصویر 7-7 نمونه‌ای از توالی رنگ خطی در فوتوشاپ

در تنظیمات مربوط به توالی رنگ فوتوشاپ مشخص است که رنگ با ماتی 40 درصد آغاز می‌شود (60 درصد شفافیت دارد) و اولین توقف رنگ نیز در موقعیت 37 درصد قرار دارد و ماتی نیز در این نقطه 0 درصد است. ما می‌توانیم از این ابزار برای جمع‌آوری داده برای استفاده در فایل CSS، بهره ببریم. نخست الگوی استاندارد W3C شروع می‌کنیم و دستورات مربوط به مرورگر فایرفاکس، اپرا 11.10 و نسخه‌های جدید مرورگرهای برپایه‌ی WebKit را می‌نویسیم:

css/styles.css (excerpt)

```
#ad2 {
  :
  background-image:
    -moz-linear-gradient(
      270deg,
      rgba(0,0,0,0.4) 0,
      rgba(0,0,0,0) 37%,
      rgba(0,0,0,0) 83%,
      rgba(0,0,0,0.06) 92%,
      rgba(0,0,0,0) 98%
```

```

);
background-image:
  -webkit-linear-gradient(
    270deg,
    rgba(0,0,0,0.4) 0,
    rgba(0,0,0,0) 37%,
    rgba(0,0,0,0) 83%,
    rgba(0,0,0,0.06) 92%,
    rgba(0,0,0,0) 98%
  );
background-image:
  -o-linear-gradient(
    270deg,
    rgba(0,0,0,0.4) 0,
    rgba(0,0,0,0) 37%,
    rgba(0,0,0,0) 83%,
    rgba(0,0,0,0.06) 92%,
    rgba(0,0,0,0) 98%
  );
}

```

ما می‌خواهیم که توالی رنگمان از بالا به سمت پایین تبلیغتمان، رسم شود از این رو زاویه را بر روی 270deg (از بالا به سمت پایین) تنظیم کردیم. سپس تمامی توقف رنگها را از فایل فوتوشاپ به کدمان منتقل کردیم. دقت کنید که ما نقطه‌ی پایانی توالی رنگ را حذف کردیم، زیرا آخرین توقف رنگ در مکان 98 درصد قرار گرفته و فاصله بین 98 درصد تا دیواره‌ی کناری عنصر، با همین رنگی که در آخرین توقف رنگ، تعریف شده بود، پر می‌شود. (در اینجا زمانی که ماتی روی 0 درصد باشد، رنگمان سیاه و در باقی موارد، شفاف کامل است.)

حال اجازه دهید که الگوی قدیمی WebKit را به کار بگیریم و در پایان نیز، الگوی استاندارد را بدون استفاده از پیشوند بنویسیم تا در آینده، مورد استفاده نسخه‌های جدید مرورگرها، قرار گیرد:

css/styles.css (excerpt)

```

#ad2 {
  :
  background-image:
    -webkit-gradient(linear,
      from(rgba(0,0,0,0.4)),
      color-stop(37%, rgba(0,0,0,0)),
      color-stop(83%, rgba(0,0,0,0)),
      color-stop(92%, rgba(0,0,0,0.16)),
      color-stop(98%, rgba(0,0,0,0)));
  background-image:
    -webkit-linear-gradient(

```

```

270deg,
rgba(0,0,0,0.4) 0,
rgba(0,0,0,0) 37%,
rgba(0,0,0,0) 83%,
rgba(0,0,0,0.06) 92%,
rgba(0,0,0,0) 98%
);
background-image:
linear-gradient(
270deg,
rgba(0,0,0,0.4) 0,
rgba(0,0,0,0) 37%,
rgba(0,0,0,0) 83%,
rgba(0,0,0,0.06) 92%,
rgba(0,0,0,0) 98%
);
}

```

حال توالی رنگ ما به خوبی در مرورگرها موزیلا (فایرفاکس)، اپرا و مرورگرهای برپایه‌ی WebKit نمایش داده می‌شود.

ساخت توالی رنگ خطی با استفاده از SVG

هنوز مرورگرهای دیگری نیز وجود دارد که باید توالی رنگ خطی را به آنها اضافه کنیم. در اپرا 11.01 و نسخه‌های قبل از آن و مهم‌تر از همه، اینترنت اکسپلورر 9 از تصاویر SVG در پس‌زمینه پشتیبانی می‌کنند. با ساخت یک توالی رنگ با فرمت SVG و قرار دادن آن به عنوان یک تصویر پس‌زمینه در این مرورگرها، می‌توانیم همان جلوه‌ای را که با استفاده از CSS3 ساخته بودیم را دوباره‌سازی کنیم.



خوب است بدانید – SVG چیست؟

SVG مخفف عبارت Scalable Vector Graphics (تصاویر برداری مقیاس‌پذیر) است. SVG زبانی برپایه‌ی XML است که از یک سری عناصر برای ساخت تصاویر برداری مقیاس‌پذیر، استفاده می‌کند. (ساختار این عناصر درست شبیه به ساختاری است که شما برای سندتان در HTML در نظر می‌گیرید.) در فصل یازدهم در مورد SVG به طور مفصل‌تر، توضیحاتی را ارائه خواهیم داد، اما اکنون تنها مقدمات را به سرعت برایتان توضیح می‌دهیم و زیرا چیزی که قصد ساختش را داریم، یک توالی رنگ ساده است.

شاید ساخت فایل SVG ترسناک به نظر برسد، اما ساخت توالی رنگ با استفاده از آن بسیار آسان و سراسر است. در این تکه کد همان توالی رنگی که در CSS3 ساخته بودیم اکنون با SVG نوشتیم:

```
images/gradient.svg (excerpt)
```



```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.0//EN" "http://www.w3.org/TR/2001/REC-SVG-20050904/DTD/svg10.dtd">
<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink"
version="1.1">
<title>Module Gradient</title>
<defs>
<linearGradient id="grad" x1="0" y1="0" x2="0" y2="100%">
<stop offset="0" stop-opacity="0.3" color-stop="#000000" />
<stop offset="0.37" stop-opacity="0" stop-color="#000000" />
<stop offset="0.83" stop-opacity="0" stop-color="#000000" />
<stop offset="0.92" stop-opacity="0.06" stop-color="#000000" />
<stop offset="0.98" stop-opacity="0" stop-color="#000000" />
</linearGradient>
</defs>
<rect x="0" y="0" width="100%" height="100%" style="fill:url(#grad)" />
</svg>
```

اگر به فایل SVG دقت کنید، متوجه خواهید شد که شباهت زیادی بین الگوی نگارش دستور توالی رنگ خطی در CSS3 و الگوی استفاده شده در SVG وجود دارد. ما نوع توالی رنگ و زاویه آن را با استفاده از عنصر `lineargradient` مشخص کرده و سپس توقف رنگ‌ها را مشخص می‌کنیم. زاویه را با مشخص کردن نقطه شروع و پایان مشخص می‌کنیم. (با تعیین مقایر `x1, y1` به `x2, y2` توقف رنگ‌ها کامل، واضح و گویا هستند و دارای خصیصه‌ای به نام `offset` که حاوی مقداری بین 0 و 1 است که موقعیت هر کدام از این توقف رنگ‌ها را مشخص می‌کند. بعد از تعریف توالی رنگ باید یک چهار ضلعی ایجاد کنیم (عنصر `rect`) و با استفاده از خصیصه‌ی `style` آن را به توالی رنگمان، بیوشانیم. حال که یک توالی رنگ جذاب و کوچک را ساختیم، چطور از آن در سایتمان استفاده کنیم؟ نخست باید فایل SVG را با پسوند `.svg` ذخیره کنید. سپس در فایل CSS خود، فایل SVG را به عنوان تصویر پس‌زمینه‌ی یک عنصر، معرفی کنید درست به همان شکل که تصاویر با فرمت‌های GIF، JPEG، و یا PNG را در پس‌زمینه قرار می‌دادید:

css/styles.css (excerpt)

```
#ad2 {
  :
  background-image: url(../images/gradient.svg);
  :
}
```

دستور مربوط به فایل SVG را باید زودتر از دستورات توالی رنگ CSS بنویسید تا به این شکل مرورگرهای جدید بتوانند هر دو را ببینند و هر مرورگری که هر دو فرمت را پشتیبانی کند، دومی را انتخاب خواهد کرد. (که در این حالت دستورات توالی رنگ CSS3 انتخاب می‌شوند.) بسیاری از

مرورگرها هوشمند عمل می‌کنند و در شرایطی که فرمت SVG در پس‌زمینه قرار گرفته باشد و پس از دستور مربوط به SVG، دستورات توالی رنگ CSS3 آمده باشد، دیگر فایل SVG را دریافت نمی‌کنند و از همان دستورات CSS3 استفاده می‌کنند.

تفاوت اصلی بین توالی رنگ در CSS و توالی رنگ در SVG در این است که تصاویر SVG که در پس‌زمینه استفاده می‌شوند، در حالت پیش‌فرض، عرض و ارتفاع 100 درصد ندارند و کل پس‌زمینه را نمی‌پوشانند. برای اینکه فایل SVG کل پس‌زمینه‌ی عنصر موردنظر ما را بپوشاند، باید خصیصه‌های width و height را در عنصر rect استفاده کنید و مقدار آن را معادل 100% قرار دهید.

توالی رنگ خطی با استفاده از فیلترهای اینترنت اکسپلورر

برای ایجاد جلوه‌ی توالی رنگ در اینترنت اکسپلورر نسخه‌های قبل از 9، می‌توانیم از فیلترهای اختصاصی IE برای ایجاد توالی رنگ‌های ساده، استفاده کنیم. فیلتر توالی رنگ در IE از توقف رنگ، پشتیبانی نمی‌کند. تنها قابلیتی که در اختیار شما قرار دارد این است که مشخص کنید، توالی رنگ به شکل افقی باشد یا عمودی و همچنین رنگ آغاز و پایان را مشخص کنید. این ساختار بسیار ابتدایی و ساده است، ولی اگر بخواهید در مرورگرهای قدیمی از توالی رنگ استفاده کنید، این فیلتر می‌تواند به کمکتان بیاید.

شیوه‌ی نگارش دستور برای IE به این شکل است:

```
filter:progid:DXImageTransform.Microsoft.gradient(GradientType=0,startColorstr='#COLOR',
endColorstr='#COLOR'); /* IE6 & IE7 */
-ms-
filter:"progid:DXImageTransform.Microsoft.gradient(GradientType=0,startColorstr='#COLOR'
, endColorstr='#COLOR)"; /* IE8 */
```

اگر بخواهید توالی رنگ به شکل افقی باشد باید پارامتر GradientType را بر روی عدد 1 و اگر بخواهید توالی رنگ، عمودی باشد، از عدد صفر تنظیم کنید. از آنجایی که توالی رنگی که ما از آن در طرحمان استفاده کرده‌ایم دارای توقف رنگ است، نمی‌توانیم از فیلتر IE استفاده کنیم. تبلیغ ما بدون توالی رنگ هم ظاهر زیبایی دارد، از این رو قصد نداریم که این توالی رنگ را به نسخه‌های قدیمی IE اضافه کنیم.



هشدار – فیلترها اثر بدی بر روی سرعت سایت دارند

همانطور که قبلاً توضیح دادیم، فیلترهای IE تأثیر شدیدی بر روی سرعت و کارایی سایت می‌گذارند، از این رو تا جایی که امکان دارد از آنها استفاده نکنید. محاسبه و نمایش جلوه‌های ایجاد شده با فیلتر، باری را بر دوش پردازنده سیستم کاربر می‌گذارد. برخی از این جلوه‌ها سریعتر و برخی کندترند. SVGها نیز اثری مشابه دارند (البته کمتر) از این رو باید سایتتان را در مرورگرهای مختلف به دقت بررسی کنید.

ابزارهای مفید

حال که شیوه‌ی ایجاد توالی رنگ خطی را فرا گرفتید با پیچیدگی‌های ساختار نگارش دستورش آشنا شدید، می‌توانید همه این مواردی که یاد گرفته‌اید را فراموش کنید. ابزارهای بسیار خوبی وجود دارند که به شما کمک می‌کنند تا توالی رنگ خطی ایجاد کنید و دیگر مجبور نیستید که دستور خود را به چند ساختار مختلف بنویسید تا در تمامی مرورگرها به درستی نمایش یابد. این ابزارها، زحمت این کار را برای شما می‌کشند.

جان آلسوپ^۱ ابزاری را طراحی کرده که به شما اجازه می‌دهد تا توالی رنگ‌های خطی به همراه توقف رنگ‌های دلخواهتان بسازید که در مرورگر فایرفاکس و مرورگرهای برپایه‌ی WebKit به درستی نمایش داده شود. این ابزار قادر است توالی رنگ خطی و حلقوی را ایجاد کند. این ابزار تنها رنگ‌های هگزادسیمال را قبول می‌کند، اما به شما اجازه می‌دهد که کد ساخته شده توسط برنامه را کپی کنید و سپس در صورت نیاز مقادیر هگزادسیمال را به RGBA و یا HSLA تغییر دهید.

دامین گالارزا^۲ ابزاری ایجاد کرده که به شما اجازه می‌دهد که تا هم از توقف رنگ استفاده کنید و همچنین رنگ‌ها را با نیز با الگوی RGB برایتان می‌نویسد. البته ابزار به شما اجازه می‌دهد که از یک جدول رنگ HSL نیز، رنگ خود را انتخاب کنید، ولی در زمان ساخت کد توالی رنگ، مقادیر HSL را به RGB تبدیل می‌کند. این ابزار، شفافیت را پشتیبانی نمی‌کند اما می‌توانید توالی رنگ دلخواهتان را با این ابزار ایجاد کنید، سپس کد RGB را ویرایش کرده و شفافیت را به آن بیافزایید. این ابزار نسبت به ابزار قبلی، امکانات بیشتری در اختیار شما قرار می‌دهد، ولی شاید محیطش برای افراد تازه کار کمی شلوغ و گیج‌کننده باشد.

در نهایت پول آیریش^۳ هم سایتی به نام css3please را راه‌اندازی کرده که به شما اجازه می‌دهد تا به وسیله‌ی آن توالی رنگ خطی بسازید اما این ابزار، از توقف رنگ، پشتیبانی نمی‌کند. شاید تعجب کنید که چرا این سایت را به شما معرفی کردیم. مایل بودیم، بدانید که این ابزار تنها ابزاری است که فیلترهای IE را هم در کنار دیگر شیوه‌های نگارش دستور توالی رنگ، برایتان ایجاد می‌کند. علاوه بر این، در این سایت می‌توانید کدهای مربوط به امکانات دیگر CSS3 همانند گوشه‌های گرد و سایه‌ها را با تمامی شیوه‌های نگارش مختلف، ایجاد کنید تا این جلوه‌ها در تمامی مرورگرها به درستی نمایش یابد.

1. <http://www.westciv.com/tools/gradients/>

2. <http://gradients.glrzad.com/>

3. <http://css3please.com/>