



Eric A. Meyer

SMASHING CSS

PROFESIONALNE TEHNIKE ZA DIZAJN SAVREMENIH WEB STRANICA



SMASHING CSS

PROFESIONALNE TEHNIKE
ZA DIZAJN SAVREMENIH
WEB STRANICA

Eric A. Meyer

 **kompiuter
biblioteka**

 **WILEY**
wiley.com

Izdavač:



**kompjuter
biblioteka**
osnovana 1986.
www.kombib.rs

Vojvode Stepe 34, Beograd

Tel: 011/3096-966

e-mail: kombib@gmail.com

internet: www.kombib.rs

Urednik: Mihailo J. Šolajić

Za izdavača, direktor:

Mihailo J. Šolajić

Autor: Eric A. Meyer

Prevod: Goran Janačković

Lektura: Zlatan Pešić

Slog : Zvonko Aleksić

Znak Kompjuter biblioteke:

Miloš Milosavljević

Štampa: „Svetlost” Čačak

CTP ploče: „Svetlost” Čačak

Tiraž: 500

Godina izdanja: 2011.

Broj knjige: 460

Izdanje: Prvo

ISBN: 978-86-7310-473-7

Smashing CSS

PROFESSIONAL TECHNIQUES FOR
MODERN LAYOUT

by Eric A. Meyer

ISBN: 978-0-470-68416-0

Copyright © 2011 Eric A. Meyer. All rights reserved.

Published by Wiley Publishing, Inc.

All right reserved. No part of this book may be reproduced or transmitted in any form or by means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Autorizovani prevod sa engleskog jezika edicije u izdanju „Wiley Publishing, Inc.”, Copyright © 2011.

Sva prava zadržana. Nije dozvoljeno da nijedan deo ove knjige bude reprodukovani ili snimljen na bilo koji način ili bilo kojim sredstvom, elektronskim ili mehaničkim, uključujući fotokopiranje, snimanje ili drugi sistem presnimavanja informacija, bez dozvole izdavača.

Zaštitni znaci

Kompjuter biblioteka i „Wiley Publishing, Inc.” su pokušali da u ovoj knjizi razgraniče sve zaštitne oznake od opisnih termina, prateći stil isticanja oznaka velikim slovima.

Autor i izdavač su učinili velike napore u pripremi ove knjige, čiji je sadržaj zasnovan na poslednjem (dostupnom) izdanju softvera. Delovi rukopisa su možda zasnovani na predizdanju softvera dobijenog od strane proizvođača. Autor i izdavač ne daju nikakve garancije u pogledu kompletnosti ili tačnosti navoda iz ove knjige, niti prihvataju ikakvu odgovornost za performanse ili gubitke, odnosno oštećenja nastala kao direktna ili indirektna posledica korišćenja informacija iz ove knjige.

CIP - Каталогизација у публикацији
Народна библиотека Србије, Београд

004.42:004.738.12
004.925

МЕЈЕР, Ерик А.

Smashing CSS: Profesionalne tehnike za dizajn
savremenih web stranica / Eric A. Meyer;
[prevod Goran Janačković]. - 1. izd. - Beograd:
Kompjuter biblioteka, 2011 (Čačak : Svetlost).
- IX, 283 str.: ilustr.; 24 cm

Tiraž 500. - O autoru: str. V. - Registar.

ISBN: 978-86-7310-473-7

a) Web презентације - Програмирање

b) Каскадне листе стилова

COBISS.SR-ID 183565836

ZAHVALNICA IZDAVAČA

Ljudi koji su omogućili da ova knjiga izađe na tržište su:

Uredništvo i produkcija

Tehnički direktor: Majkl Liti (Michelle Leete)

Pomoćni direktor – upravljanje sadržajem knjige: Martin Trajb (Martin Tribe)

Izdavač saradnik: Kris Veb (Chris Webb)

Pomoćnik za izdavanje: Eli Skot (Ellie Scott)

Urednik projekta: Brajan Herman (Brian Herrmann)

Urednik umnožavanja: Debi Batler (Debbye Butler)

Urednički menadžer: Džodi Džensen (Jodi Jensen)

Urednik projekta: Sara Šler (Sara Shlaer)

Asistent uredništva: Lesli Saksman (Leslie Saxman)

Marketing

Vodeći marketing menadžer: Luiz Brajnholt (Louise Breinholt)

Izvršni direktor marketinga: Kejt Peret (Kate Parrett)

Usluge sastavljanja

Sastavljač: Dženifer Mejberi (Jennifer Mayberry)

Lektor: Suzan Hobs (Susan Hobbs)

Indekser: Potomac Indexing, LLC

ZAHVALNICE

Zahvaljujem se Krisu Vebu (Chris Webb) što me je odveo u inostranstvo, a zatim pružio podršku u svim kašnjenjima, neuspesima i posrtanjima. U nekim trenucima izgledalo je kao da projekat neće biti nastavljen, a Kris je svaki put davao podršku za nastavljanje sa radom, svojim humorom i beskonačnom smirenošću. Moje puno poštovanje, a nadam se da ćemo nekada biit u mogućnosti da istovremeno odemo na odmor, pa da zajedno koktelima nazdravljamo pored bazena.

Zahvalnicu dugijem i Debi Batler (Debbye Butler) i Brajanu Hermanu (Brian Herrmann), koji su me vodili kroz uređivački proces i isticali mesta na kojima sam pravio greške, davao nedovoljno jasna objašnjenja, odnosno, generalno, bio nejasan.


Zahvaljujem se svima koji su čitali ono što sam napisao, bilo na papiru ili na web stranicama: hvala vam, hvala vam, hiljadu puta vam hvala na svemu.

Posebnu zahvalnost dugujem svojoj ženi i ćerkama, mnogo više nego što se može rečima opisati.

Erik A. Mejer
(Eric A. Meyer)
Cleveland Heights, Ohio
13. Avgust 2010. godine

O AUTORU

ERIK MEJER (ERIC MEYER) je svetski priznati ekspert za HTML, CSS i Web standarde, a radi na Webu još od 1993. godine. Osnivač je Complex Spiral Consulting kompanije, čiji su klijenti America On-Line, Apple, Adobe, Microsoft Progressive Insurance, Sherwin-Williams i brojne druge velike kompanije; suosnivač je pokreta o mikroformatima i zajedno sa Džefrijem Zeldmanom (Jeffrey Zeldman) An Event Apart serije konferencija namenjenih ljudima koji prave web stranice. Autor je pet veoma prodavanih knjiga o CSS kodu i web dizajnu.



*Ova knjiga je posvećena
Ketrin, Kerolin i Rebeki.*

SADRŽAJ

DEO I OSNOVE.....3

Poglavlje 1 Alati 5

Firebug	6
Web Developer paleta sa alatkama	13
Internet Explorer Developer paleta da alatkama (ili alati)	17
Dragonfly (za web čitač Opera)	22
Web Inspector (Safari čitač)	26
XRAY alat.....	29
SelectORacle alat	30
Dijagnostički stilovi.....	31
Reboot stilovi	33
IE9.js skript	36

Poglavlje 2 Selektori 39

Pseudo šta?	40
Odredišta sa stilom	41
Specifičnost	43
Važnost	44
Šta se događa ukoliko izostavite ključne reči za skraćeno definisanje vrednosti	45
Selektivno predefinisane skraćeno definisanih svojstava	47
Univerzalna selekcija	49
Id i class atribut	50
Id i class atribut	52
Korišćenje više class atributa	53
Selekcija jednostavnih atributa	54
Klase i selektovanje atributa	56
Identifikator i selektor atributa	57
Selektovanje podstringa atributa	58
Dodatno selektovanje podstringa atributa	60

Selektovanje izvedenog elementa.....	62
Simulirana parcijalna selekcija naslednika	63
Selektovanje srodnih elemenata	65
Generisani sadržaj	67

DEO II SUŠTINA71

Poglavlje 3 Saveti73

Validacija!.....	74
Definisanje redosleda font vrednosti	75
Promena line-height vrednosti.....	75
Bezdimenzione line-height vrednosti	76
Izbegavanje border vrednosti bez definisanog stila	78
Upravljanje izgledom ivica pomoću boja.....	78
Sprečavanje prikazivanja elementa	80
Promena vidljivosti elementa	82
Uklanjanje elemenata sa ekrana.....	83
Zamena teksta slikom.....	84
Stilovi štampanja	87
Razvoj stilova štampanja	87
Linkovi na nivou blokova.....	89
Definisanje margine ili pomeranje teksta?.....	90
Pomeranje stavki listi prema marginama	92
Postavljanje markera u liste	93
Markeri u pozadini	95
Generisanje markera	98
Postoji više kontejnera nego što mislite.....	100
Pozadine dokumenta	103
CSS kod specifičan za server.....	104

Poglavlje 4 Raspored elemenata107

Konture umesto ivica	108
Centriranje blokova	110
Suzbijanje plivanja elementa: prekoračenje.....	113
Suzbijanje plivanja elementa: pomeranje	115
Clearfix tehnika	117
Uklanjanje susednog elementa.....	118
Dve jednostavne kolone	120
Tri jednostavne kolone	121
Lažne kolone	124
Prilagođavanje promeni veličine prozora.....	127
Pravi raspored elemenata.....	131
Sveti gral	135
Promenljive mreže	140
Raspored zasnovan na em vrednostima.....	144
Negativne margine u toku dokumenta.....	147

Pozicioniranje u okviru konteksta	150
Uklanjanje bloka koji sadrži elemente.....	152
Fiksirana zaglavlja i futeri.....	155
Poglavlje 5 Efekti	159
Kompleksna spirala	160
CSS iskačući elementi	165
CSS meniji.....	167
Uklanjanje delova boksa	169
Zaobljene ivice u verzijama starijim od CSS 3.....	172
CSS 3 i zaobljeni uglovi	177
CSS sprajtovi.....	179
Klizanje vrata	182
Modifikovano klizanje vrata	185
CSS paralaksa	187
Postavljanje teksta pored ivica nepravilnog oblika	190
Preciznije postavljanje teksta pored ivica nepravilnog oblika	194
Postavljanje slika u okvir	198
Ograničene slike	200
DEO III NAPREDNE TEHNIKE	203
Poglavlje 6 Tabele	205
Zaglavlje, telo, futer	206
Zaglavlja vrsta	208
Definisanje stilova na nivou kolona.....	210
Prikazivanje podataka iz tabele na mapi	215
Grafici dobijeni na osnovu tabela	222
Poglavlje 7 (Bliska) budućnost.....	231
Stilizovanje korišćenjem HTML 5 koda	232
Definisanje klasa u HTML 5 kodu.....	234
Upiti vezani za medije.....	235
Definisanje stila za određene izvedene elemente	240
Definisanje stila za proizvoljne kolone	243
HSL i HSL alfa boja.....	248
Stilovi za definisanje senki	250
Višestruke pozadine	251
Dvodimenzionalne transformacije	256
INDEKS	269



UVOD

CSS je postao izuzetno uspešan – gotovo podjednako uspešan kao i HTML – a to je ponekad teško приметiti. On se sada primenjuje svuda, počevši od web pregledača do skladišta aplikacija i klijenata za interaktivno komuniciranje, a pri tome ne pokazuje bilo kakve znake laganog nestajanja. Kako se oblasti primene jezika proširuju, tako njegove mogućnosti i dalje napreduju.

Ova knjiga sadrži skoro 100 saveta, tehnika, alata i trikova za pravljenje izuzetno privlačnih web stranica korišćenjem CSS jezika. Svako objašnjenje je potpuno nezavisno od ostalih: slučajnim izborom možete da pristupite određenoj stranici u knjizi, a zatim pročitate sadržaj, a da se pri tome ne brinete da ste nešto veoma značajno propustili preskakanjem sadržaja u odgovarajućem poglavlju. To znači da je sadržaj knjige takav da podrazumeva da imate određeno znanje vezano za CSS jezik i način njegovog korišćenja u praksi. Pretpostavljeni nivo znanja za razumevanje se može najbolje opisati kao „početni/srednji nivo“. To znači da, ukoliko ste početnik, ili znate o CSS jeziku više od ljudi koji pišu specifikacije, verovatno nećete imati mnogo koristi od ove knjige. Svi ostali mogu mnogo toga da nauče i da uživaju čitajući ovu knjigu.

U prvom delu ove knjige prikazan je pregled korisnih alata i osnovnih tehnika, uključujući i neke manje jasne CSS selektore. Drugi deo prikazuje šta možete da uradite koristeći CSS jezik, uključujući interesantne efekte, različite načine za postizanje istog rezultata, rasporede elemenata i još mnogo toga. Nakon toga, u trećem delu su prikazane napredne i najsavremenije tehnike, koje verovatno još nisu pogodne za primenu u svakom projektu na kome trenutno radite, ali će u bliskoj budućnosti postati osnova vašeg rada.

Na web stranici posvećenoj ovoj knjizi, na adresi www.wiley.com/go/smashingcss, možete preuzeti primere koda.

Pre nešto manje od jedne decenije vam je možda oprošteno što ste pomišljali da je CSS jezik na kraju svog životnog ciklusa, ali u ovoj deceniji on i dalje postoji i primenljiviji je nego ikada do sada. Nadamo se da ćete uživati u čitanju materijala koji se nalazi u ovoj knjizi, kao i u pisanju CSS koda!





OSNOVE

Poglavlje 1: Alati

Poglavlje 2: Selektori



1

ALATI

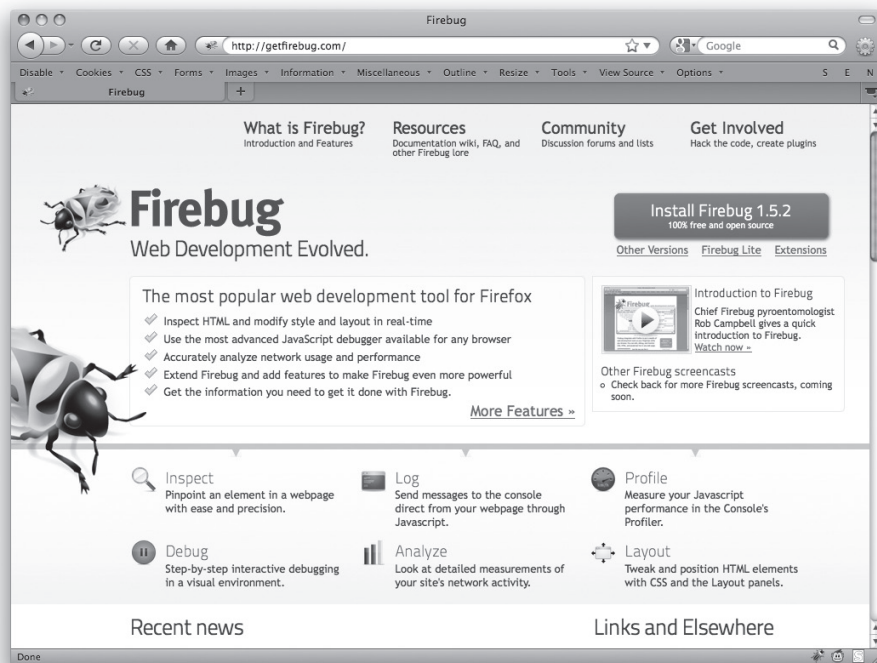
PROCES KREIRANJA web stranica (ili web aplikacija), kao i ostali poslovi koji se obavljaju na računaru, olakšan je korišćenjem raznih alata. Kada se radi o CSS kodu, postoje alati koji omogućavaju jednostavno pisanje CSS koda i korišćenje tog koda. Postoje i alati

koji omogućavaju proširivanje postojećih mogućnosti web čitača sa stanovišta mogućnosti primene CSS koda.

Ukoliko ste programer, korisnik alata ili projektant, u ovom poglavlju možete pronaći detalje koji će vam biti od velike pomoći pri svakodnevnom radu.

FIREBUG

Firebug (slika 1-1) je jedan od najosnovnijih alata za svakoga ko se bavi kreiranjem web aplikacija (drugi najosnovniji alat opisan je u odeljku “Web Developer Toolbar”). To je potpuno besplatna ekstenzija za Firefox čitač. Ukoliko koristite neki drugi web čitač, nastavite sa čitanjem: Možete, takođe, da vidite Firebug u akciji!



SLIKA 1-1: Firebug početna stranica.

Da biste preuzeli kopiju na lokalni računar, neophodno je da pristupite web stranici koja se nalazi na adresi getfirebug.com. Kliknite Install taster (u trenutku dok ovo pišemo nalazi se u gornjem desnom uglu), a zatim omogućite da se instalira. Ponovo pokrenite Firefox i budite spremni na iznenađenje.

Nemoguće je u jednom savetu opisati sve ono što nudi Firebug; zapravo, ni celo jedno poglavlje ne bi bilo dovoljno za to. Slede neke najosnovnije napomene.

Na levoj strani HTML kartice (slika 1-2) prikazana je hijerarhijska struktura vašeg dokumenta (sa podebljanim strelicama koje prikazuju ili uklanjanju podstablo dokumenta). Obratite pažnju na to da ukoliko postavite pokazivač miša na naziv elementa u HTML kartici, taj element postaje posebno naglašen na samoj stranici. To omogućava da se prikazuje oblast sa sadržajem, ali i pomeraji i margine za pojedinačne elemente pomoću posebno obojenih regiona, što je fantastično. U trenutku dok ovo pišemo, oblast sa sadržajem prikazuje se svetlo plavom bojom, pomeraji svetlo ljubičastom bojom, dok su margine označene žutom bojom, ali same boje nisu toliko značajne koliko činjenica da ih možete videti neposredno na stranici.



SLIKA 1-2: Vizuelizacija elemenata kojim se definiše raspored pomoću Firebug alata.

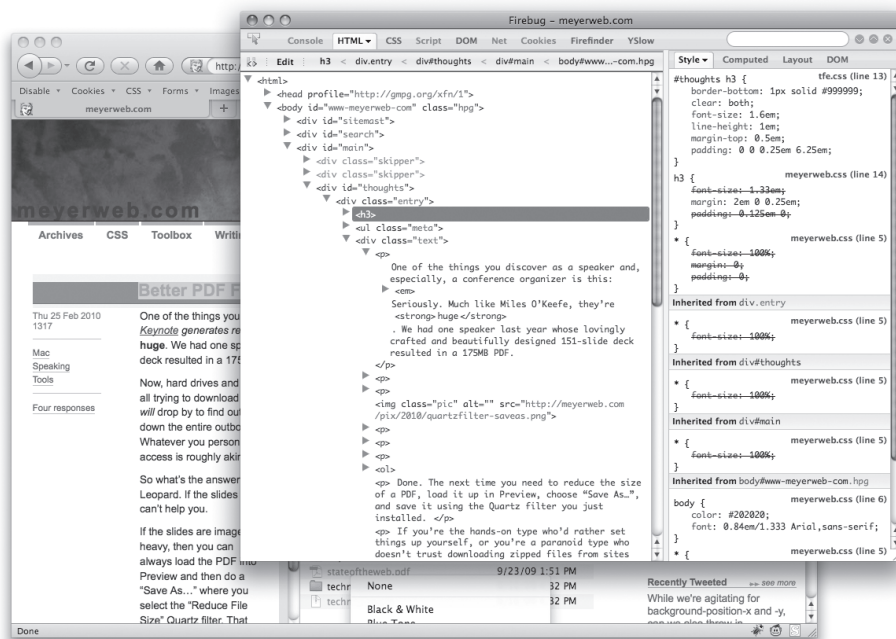
Sa desne strane HTML kartice možete da primetite CSS kod koji se primenjuje na trenutno razmatrani element, i to tako što kliknete Style karticu (slika 1-3). To nisu samo materijali koje je autor definisao, već i ono što sam web čitač primenjuje na osnovu ugrađenih stilova. Na primer, ukoliko primetite stilove kao što su `html.css` ili `quirk.css`, to su ugrađeni stilovi. (Ovi stilovi se nazivaju “UA stilovi”, odnosno user agent stilovi. Da li će se ovi stilovi primenjivati, možete da podešavate u meniju na Style kartici.)

Dešava se, često, da Firebug prikazuje i svojstva koja niste definisali, kao što je `-moz-background-clip`. Ukoliko ste sigurni da ih niste eksplicitno deklarirali, najčešće ih možete ignorisati. Takođe, ukoliko primenjujete skraćeno definisano svojstvo, ono će biti prošireno tako da se prikazuju pojedinačna svojstva. Na primer, skraćeno definisano svojstvo je:

```
font: 1em "Andale Mono", "Courier New", Courier, monospace;
```

...koje će Firebug nakon proširivanja prikazati na sledeći način:

```
font-family: "Andale Mono","Courier New",Courier, monospace;
font-size: 1em;
font-size-adjust: none;
font-stretch: normal;
font-style: normal;
font-variant: normal;
font-weight: normal;
line-height: normal;
```



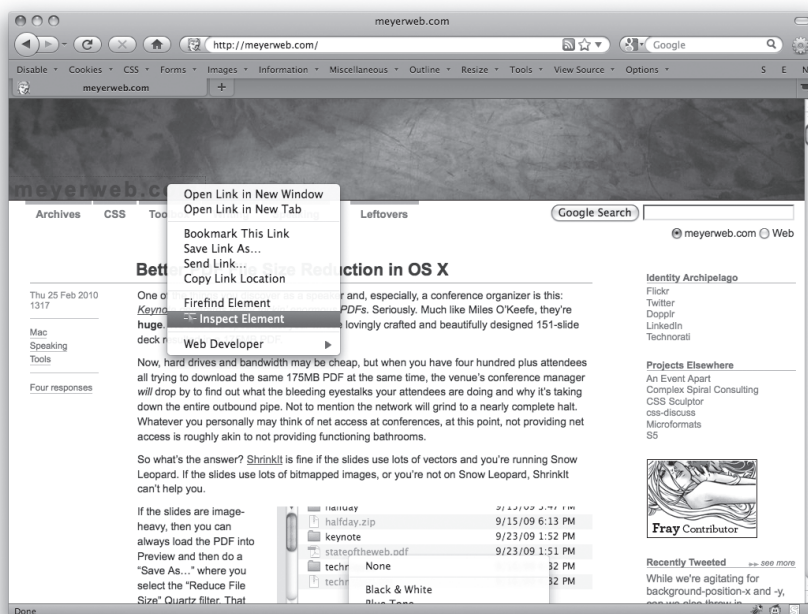
SLIKA 1-3: Firebug otvara poseban prozor za prikazivanje sadržaja Style kartice.

Iako ovaj način prikazivanja nije loš – podseća vas na to da skraćeno definisanje svojstava u sebi sadrži mnogo više nego što to izgleda na prvi pogled – može dovesti do zabune. (Više informacija o skraćeno definisanim svojstvima možete pronaći u Poglavlju 2.)

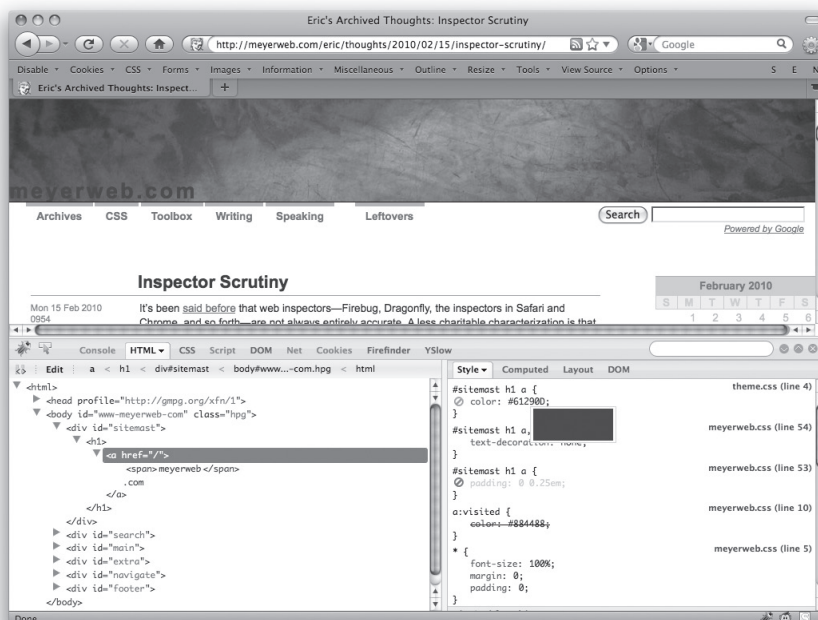
Ono što je važno napomenuti je da su pravila na Style kartici prikazana u obrnutom redosledu prema specifičnosti; prvo su navedena najspecifičnija pravila vezana za posmatrani element, zatim manje specifična, i tako redom. (Više informacija o specifičnosti možete pronaći u odeljku „Specifičnost” u Poglavlju 2.)

Veoma jednostavno možete da pregledate proizvoljni element, tako što ga kliknete desnim tasterom miša i iz kontekstnog menija, koji se pri tome otvara, selektujete Inspect Element stavku (slika 1-4). Takođe, možete da kliknete malu ikonu u obliku pokazivača, kako biste prešli u režim inspekcije. Pristupanjem različitim elementima stranice, selektovaće se sami elementi, a ukoliko ih kliknete, prikazivaće se njihov sadržaj.

Ukoliko kliknete levo od bilo koje deklaracije možete je onemogućiti u Firebug alatu. To je veoma korisno kada pokušavate da utvrdite kako svojstva interaguju testiranjem efekata isključivanja jednog od svojstava. Kao što je prikazano na slici 1-5, možete da vidite i malo polje određene boje, koja je određena na osnovu deklaracije boje, tako što postavite pokazivač miša iznad odgovarajuće vrednosti.



SLIKA 1-4: Opcije u Inspect Element kontekstnom meniju.



SLIKA 1-5: Onemogućeni stilovi i prikazivanje polja bojom definisanom specifikacijom.

Firebug Style kartica može da prikazuje i stilove elemenata koji su definisani izračunavanjem (slika 1-6), što znači da će se prikazivati vrednosti koje web čitač primenjuje za svako CSS svojstvo koje identifikuje, bilo da je to programski definisano ili ne. Zapamtite, sva CSS svojstva imaju svoje podrazumevane vrednosti; ovde možete da ih pregledate. To može biti veoma korisno ukoliko, na primer, želite da znate tačno koliko piksela web pregledač primenjuje za `line-height` prilikom prikazivanja zaglavlja.

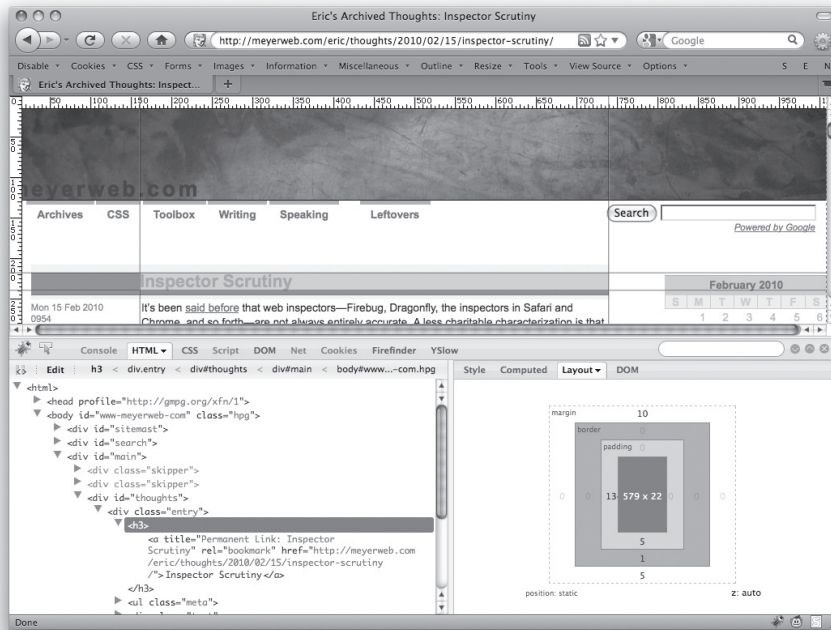


SLIKA 1-6: Izračunati stilovi.

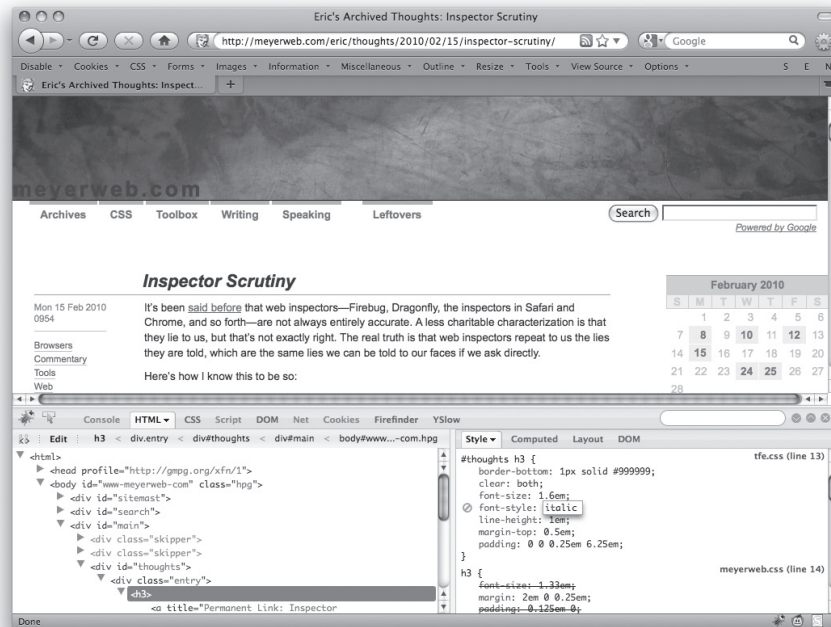
Takođe, možete da pregledate tačne dimenzije i veličine pojedinačnih elemenata, kao što su visina, širina, pomeraj, margine i ostalo (slika 1-7). Ove veličine prikazane su u pikselima. Još bolje: ukoliko postavite pokazivač miša iznad polja prikazanog u ovom panelu, prikazuju se lenjiri za piksele na samoj stranici, i to na gornjoj i levoj ivici samog elementa.

Postoji još mnogo toga: kao dokaz, na slici 1-8 možete da ažurirate vrednosti atributa elemenata (kao što je `class`) ili sam sadržaj elemenata, dodajete ili ažurirate CSS svojstva i vrednosti, i još mnogo toga.

Interfejs pregledate tako što ga kliknete ili pritisnete desni taster miša u okviru Firebug interfejsa, nakon čega možete da vidite koje operacije vam stoje na raspolaganju.



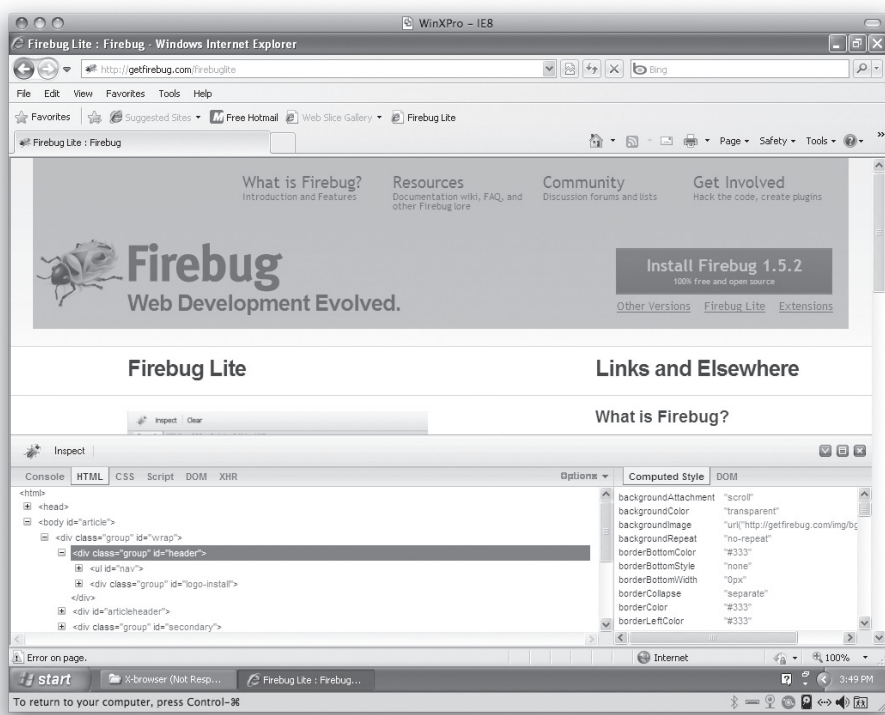
SLIKA 1-7: Layout kartica.



SLIKA 1-8: Ažuriranje CSS koda u letu u okviru Style kartice.

Jedno upozorenje: Prilikom pregledanja CSS koda elementa u Style kartici, nećete imati mogućnost da vidite pravila upravljana pseudoelementima, koja utiču na posmatrani element. Na primer, ukoliko imate pravilo koje koristi selektor `p:first-letter`, to pravilo se neće prikazivati prilikom pregledanja `p` elementa. Pseudo-klase će se prikazivati, ali pseudo elementi se ne prikazuju. To može biti posebno izazovno ukoliko koristite automatsko rešenje koje uključuje generisani sadržaj (više informacija o tome možete pronaći u odeljku „Automatsko generisanje sadržaja“ u Poglavlju 4).

Ukoliko ne koristite Firefox za razvoj, već želite da koristite osnovne blagodeti Firebug alata, pristupite web stranici koja se nalazi na adresi getfirebug.com/lite.html (slika 1-9), a zatim sledite instrukcije za omogućavanje korišćenja ovog alata u web čitačima Internet Explorer, Opera ili Safari, zavisno od toga koji od prethodno navedenih programa koristite. Možete da pristupite stranici koju testirate, ili da je dodate u listu omiljenih stranica kao bookmarklet (što vam preporučujemo da učinite).



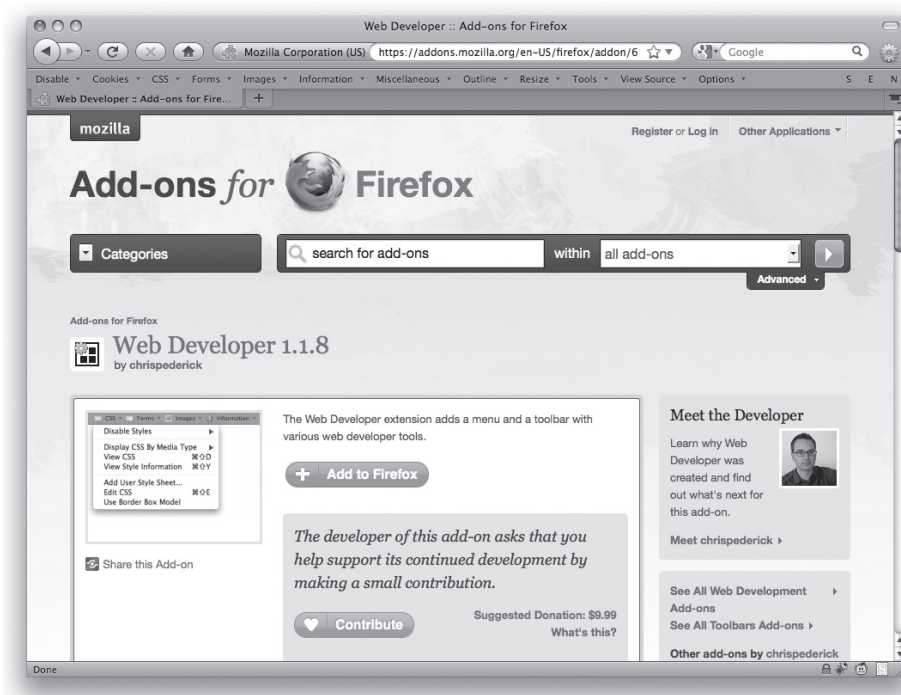
SLIKA 1-9: Firebug Lite se izvršava u okviru Internet Explorer čitača.

Ova verzija Firebug alata ne sadrži sve opcije koje postoje u Firefox ekstenziji – zato u nazivu i postoji „Lite“ – ali i dalje predstavlja veoma funkcionalan i koristan alat.

WEB DEVELOPER PALETA SA ALATKAMA

Pored Firebug alata, Web Developer Toolbar (WDT) je još jedan od osnovnih alata koje koriste autori web stranica. To je potpuno besplatna ekstenzija za Firefox.

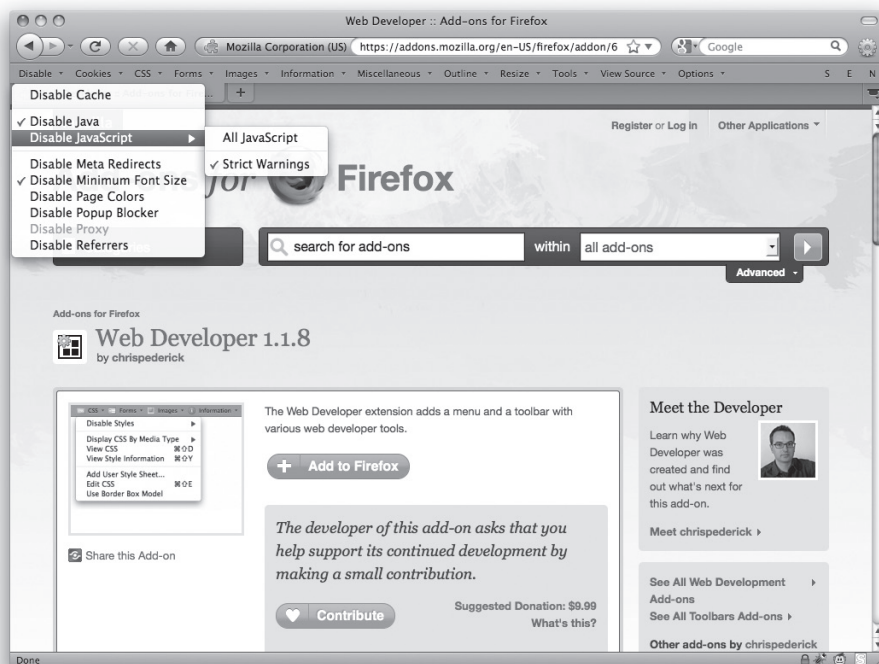
Da biste preuzeli kopiju, neophodno je da pristupite web stranici koja se nalazi na adresi chrispederick.com/work/web-developer i da instalirate dodatak. Alternativno, možete da pristupite web stranici na adresi addons.mozilla.org, unesete ključne reči Web Developer Toolbar, a zatim instalirate dodatak sa odgovarajuće WDT stranice (slika 1-10).



SLIKA 1-10: Web Developer toolbar stranica na adresi addons.mozilla.org.

Isto što važi za Firebug, važi i za WDT – ne postoji način da se opiše sve ono što vam pruža WDT; zapravo, celo poglavlje nije dovoljno da se opišu funkcije ovog alata. Navešćemo samo osnovne funkcije koje su date u određenim menijima, ali je na vama da nakon instaliranja WDT alata posvetite dodatno vreme za pregledanje svih menija i opcija koje vam stoje na raspolaganju.

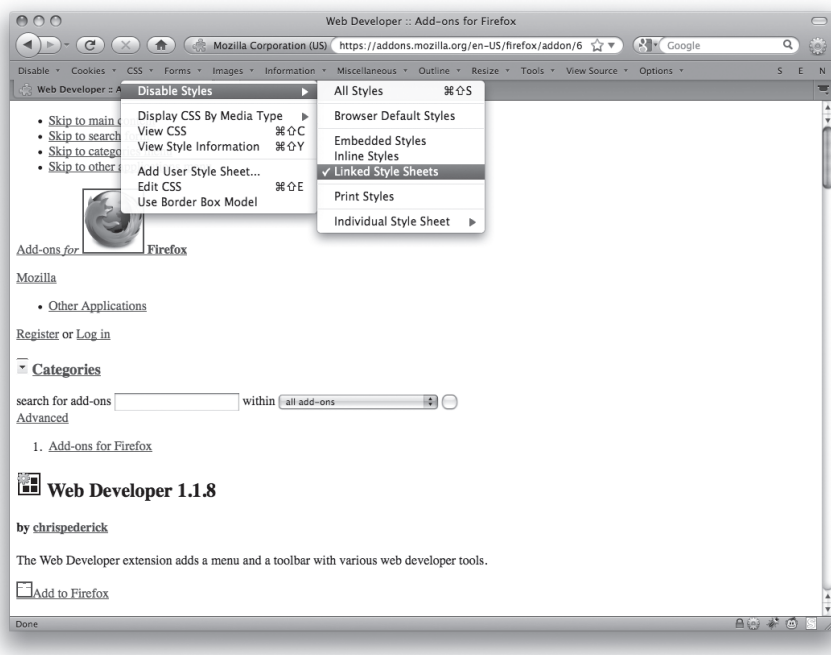
Možete da onemogućite kreiranje stranice, što je korisno ukoliko obavljate veliki broj manjih izmena, a keš web čitača postane nedovoljno ažuran čuvajući prethodne verzije stranice. Takođe, možete da isključite i JavaScript (slika 1-11), koji je koristan prilikom utvrđivanja šta se dogodilo sa stranicom prilikom nemogućnosti izvršavanja skripta, ili nemogućnosti učitavanja JavaScript radnog okvira.



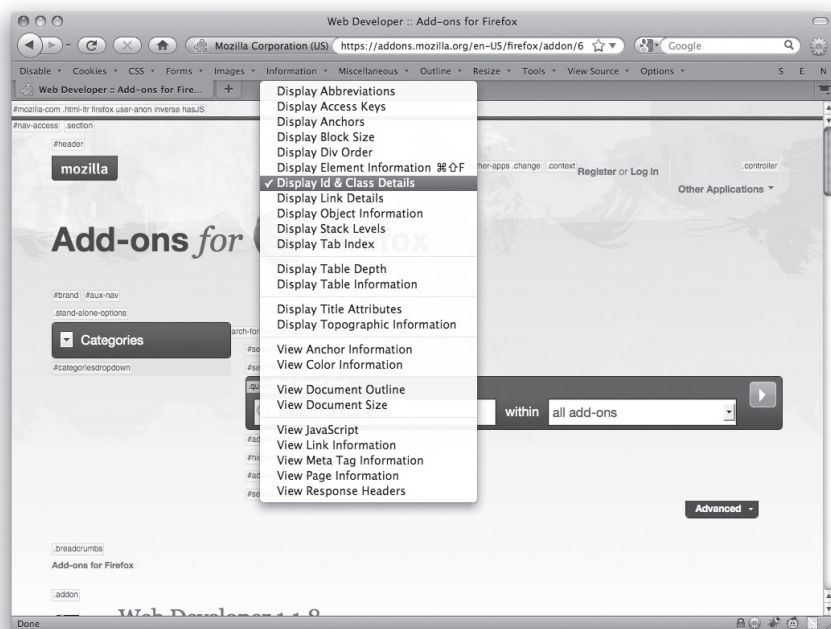
SLIKA 1-11: Stavke Disable menija.

Nekoliko stavki u CSS meniju su opcije koje omogućava i Firebug alat, ali jedna veoma korisna mogućnost je isključivanje ugrađenih stilova ili samo povezanih stilova (kao što je prikazano na slici 1-12), ili linijskih stilova. (To ne znači da treba da koristite linijske stilove!) Možete da uklonite i većinu stilova koji su definisani za korišćeni web čitač, ukoliko želite da vidite kako sadržaj može da izgleda neugledno.

Information meni (slika 1-13) sadrži veliki broj interesantnih opcija, uključujući prikazivanje informacija o klasi i identifikatorima u samom dokumentu; praćenje div redosleda stranice; pregled boja koje se primenjuju na stranici; i još mnogo toga. Možete da primenjujete režim za prikazivanje informacija o elementima, tako što selektujete proizvoljni element, nakon čega dobijate pregled svih njegovih atributa i njihovih vrednosti, njegovu poziciju na stranici, informacije o korišćenom fontu, prethodnim i narednim elementima i ostalo. Information meni je veoma sličan sa XRAY, sa stanovišta informacija koje vam pruža; više informacija vezanih za XRAY možete pronaći kasnije u toku ovog poglavlja.

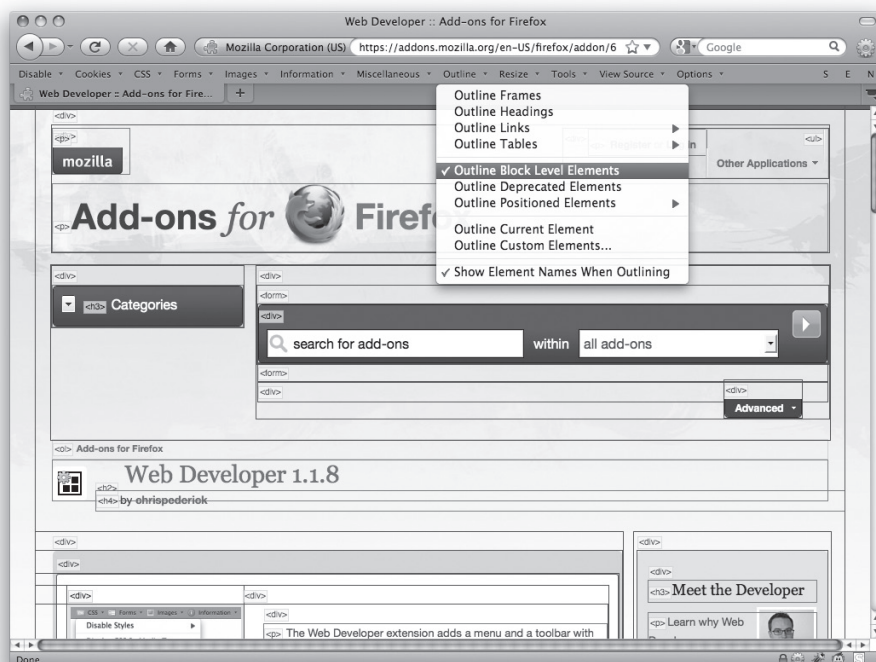


SLIKA 1-12: Onemogućavanje povezanih stilova prikazivanja pomoću CSS menija.



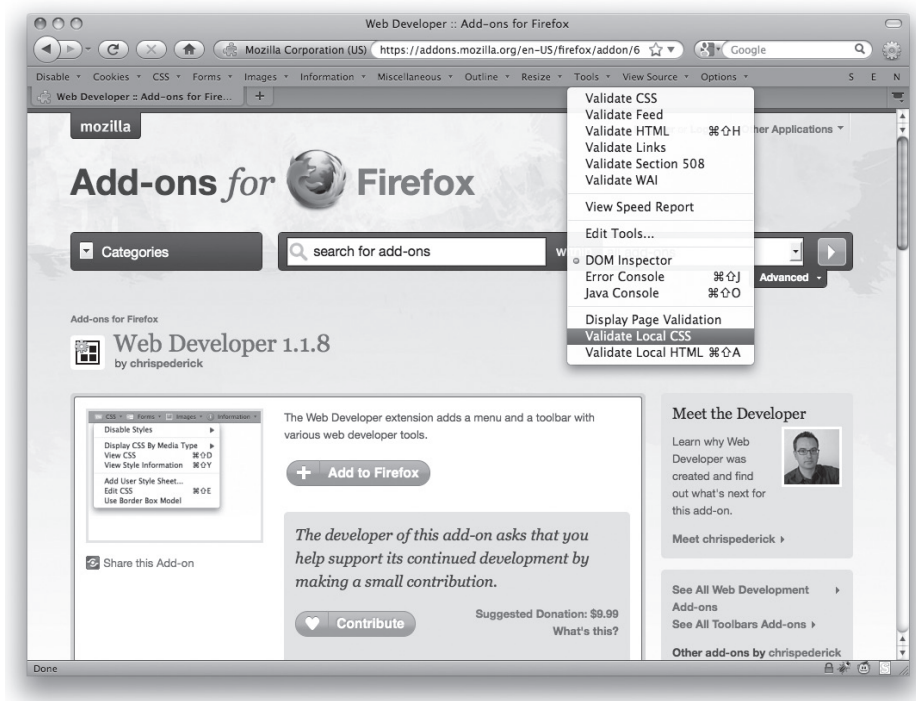
SLIKA 1-13: Prikazivanje vrednosti klase i identifikatora pomoću Information menija.

U okviru Outline menija, koji je prikazan na slici 1-14, možete da označite klase elemenata – sve blokove, sve linijske elemente, sve linkove, sve pozicionirane elemente, sve ćelije u tablici, i tako dalje – kao i da podesite samostalno definisane kolekcije elemenata i boja. Možete da izaberete i da li će se nazivi elemenata prikazivati na stranici nakon što ih označite. Ovo je mnogo korisniji meni nego što to izgleda na prvi pogled. On deluje kao meni za dijagnostikovanje rasporeda elemenata, jer veoma brzo možete da pregledate raspored skupova elemenata i utvrđujete njihov raspored u odnosu na ostale elemente, odnosno da identifikujete gde može da se javi potencijalni problem.



SLIKA 1-14: Označavanje i identifikovanje blokova korišćenjem Outline menija.

Tools meni omogućava brz pristup velikom broju validatora, funkcija za proveru postojanja grešaka i konzola za debugovanje. Jedna od najboljih funkcionalnosti je prisustvo Validate Local HTML i Validate Local CSS stavki (slika 1-15). U obe situacije, stranica koju trenutno pregledate je upakovana u serijalizovane stringove i prosleđena do relevantnog validatora. Tako, ukoliko selektujete Validate Local HTML, kod stranice se prosleđuje do HTML validatora, a vi ćete dobiti izveštaj o tome. Ovo je veoma korisno prilikom validacije stranica koje se prikazuju iza zaštitnog zida, ili sa hard diska vašeg računara – odnosno, stranica koje nisu na javno dostupne na Webu, pa samim tim im i ne može pristupiti servis za validaciju. Usled postojanja ove funkcionalnosti, lokalna validacija više ne predstavlja nikakav problem.



SLIKA 1-15: Tools meni.

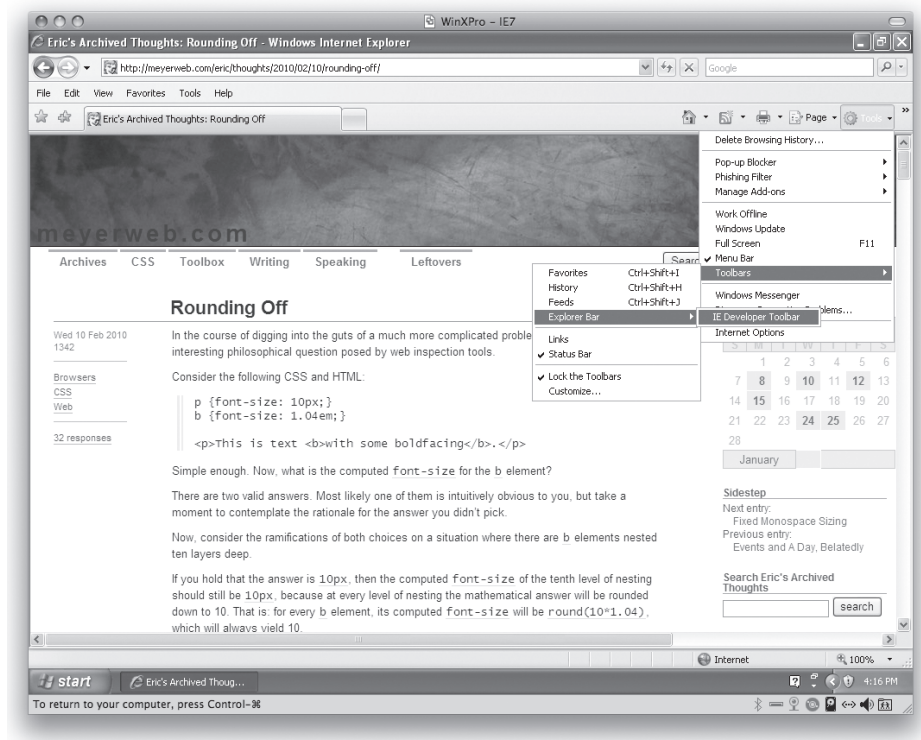
Kao što je napisano na početku ovog odeljka, ovo su samo osnovne WDT funkcionalnosti, a neophodno je uložiti dosta vremena i truda da bi se pregledali svi alati koji vam olakšavaju posao.

INTERNET EXPLORER DEVELOPER PALETA DA ALATKAMA (ILI ALATI)

Ukoliko prevashodno koristite Internet Explorer 7 prilikom razvoja web aplikacija, onda niste u mogućnosti da instalirate Web Developer Toolbar (videti prethodni odeljak). Umesto toga, možete da instalirate Internet Explorer Developer Toolbar (IEDT).

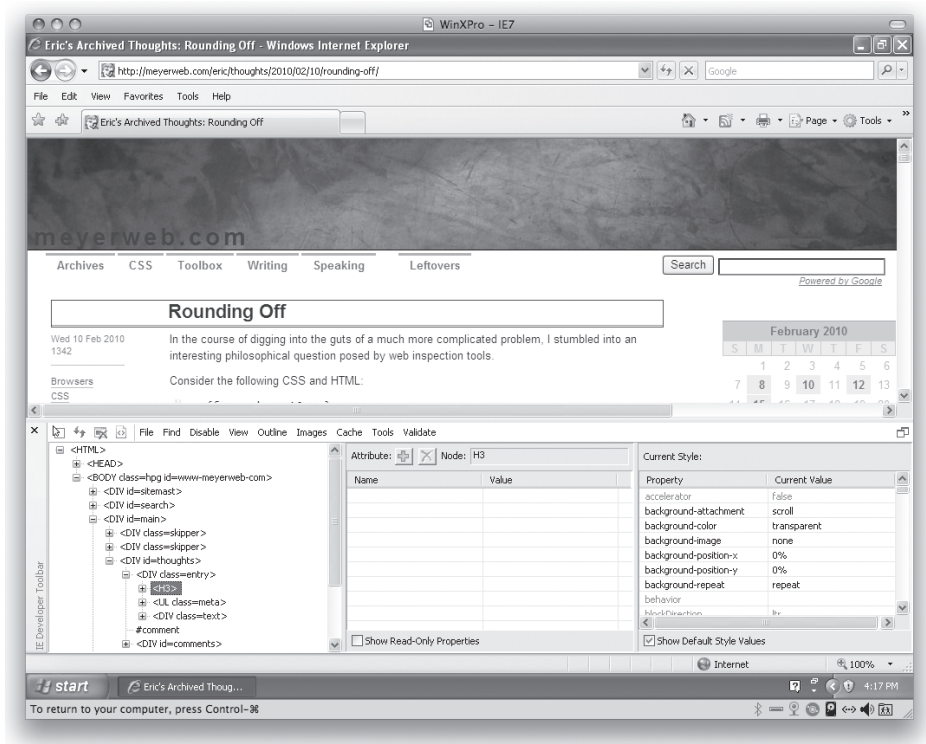
URL adresa za IEDT je jedna od klasičnih Microsoft URL adresa koje je nemoguće dešifrovati, zbog čega je najjednostavnije da pristupite svom omiljenom pretraživaču (možete da koristite Google da bi ironija bila veća), a zatim ukucajte ključne reči **Internet Explorer Developer Toolbar**. Prvi rezultat pretraživanja bi trebalo da bude željena adresa. Pristupite web stranici na toj adresi, a zatim instalirajte alat ukoliko koristite IE7. IEDT ne funkcioniše u IE8; ono što je neophodno za rad u IE8 biće opisano u daljem tekstu.

Nakon što završite instalaciju, pristupite Tools meniju, koji se nalazi u gornjem desnom uglu vašeg web čitača, pored Pages menija (videti sliku 1-16), a ne Tools meniju koji se nalazi levo između stavki Favorites i Help. U tom meniju izaberite Toolbars, zatim Explorer Bar, a nakon toga (konačno!) IE Developer Toolbar.



SLIKA 1-16: Pronalaženje IE Developer paleti sa alatka u IE7.

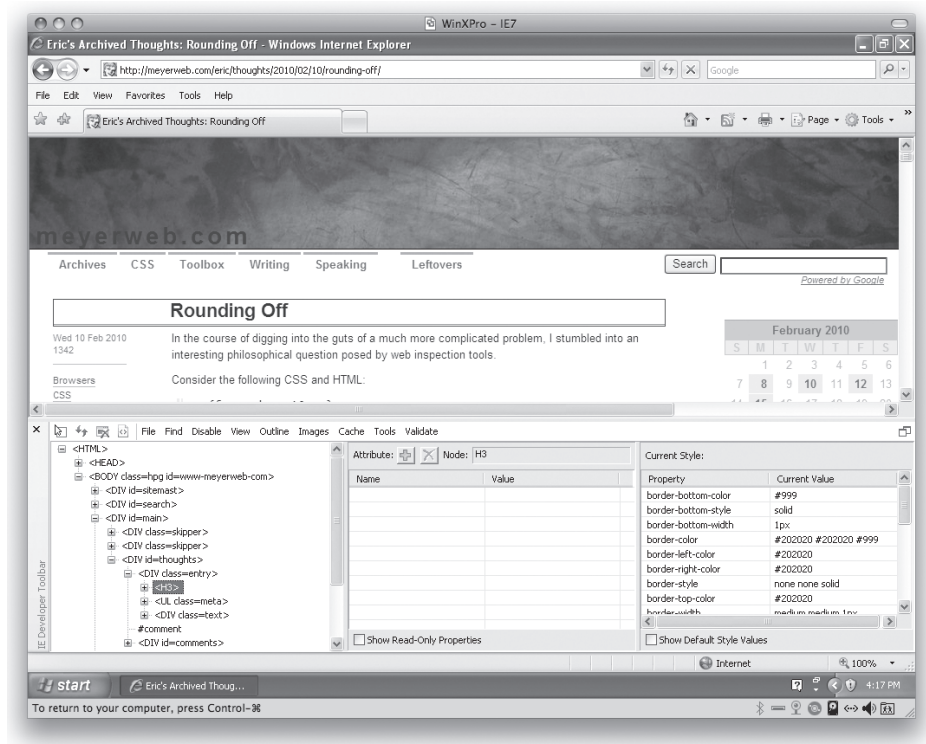
Nakon što pristupite IE Developer paleti sa alatka, u donjem delu prozora web pregledača prikazuje se panel sličan Firebug alatu, kao što je prikazano na slici 1-17. U gornjem delu panela postoje i meniji slični onima u Web Developer paleti sa alatka. Možete da postavite sve u samostalni prozor, tako što kliknete malu “two windows” ikonu koja se nalazi u gornjem desnom delu panela, pored close tastera. Proširivanje panela na taj način može biti posebno korisno ukoliko imate na raspolaganju računar sa niskom rezolucijom, kao što je ekran prenosnog računara, ili prikazujete sliku pomoću projektora.



SLIKA 1-17: IE Developer paleta sa alatkama u akciji.

Jedna veoma korisna funkcionalnost ove palete sa alatkama je mogućnost izbora između prikazivanja izračunatih stilova i deklariranih stilova, selektovanjem Show Default Style Values polja za potvrdu (slika 1-18).

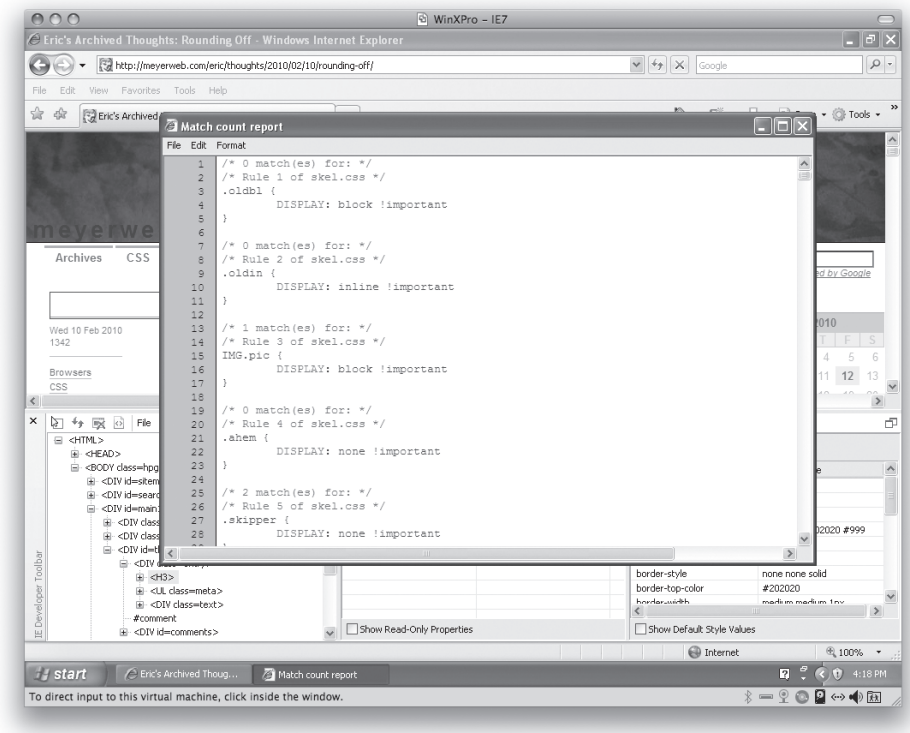
Slično, možete da koristite Show Read-Only Properties kako bi se prikazivali svi aspekti DOM (Document Object Model) svojstava posmatranog elementa. Ukoliko niste dovoljno vešti u pisanju JavaScript i DOM skriptova, ova opcija vam nije neophodna. (Nama svakako nije.)



SLIKA 1-18: IE Developer paleta sa alatima sa onemogućenom Show Default Styles opcijom.

IE Developer Toolbar sadrži podskup Web Developer Toolbar funkcionalnosti, ali najveći broj zaista korisnih alata, kao što je selektovanje elemenata i lokalna validacija HTML i CSS koda, nalaze se u prvom sloju. Postoji i izvanredna stavka pod nazivom

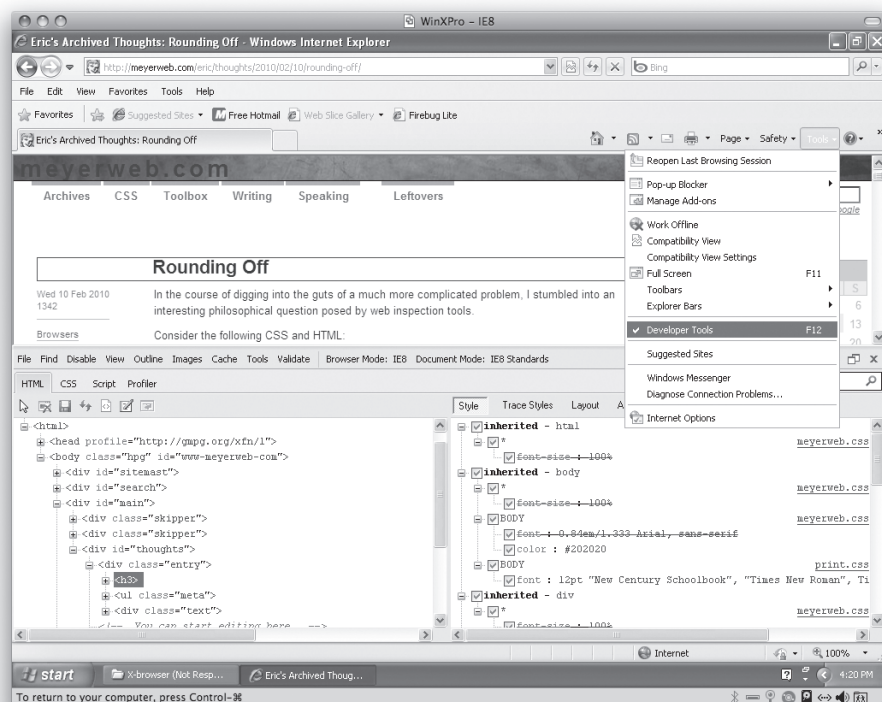
CSS Selector matches (videti sliku 1-19). Ona omogućava otvaranje prozora u kome se prikazuju sva pravila u CSS kodu i koliko puta se svaki element pojavljuje u dokumentu. Sva pravila u kojima stoji tekst „0 match(es) for:“ ne pojavljuju se na stranici, tako da možete da razmotrite uklanjanje tih pravila.



SLIKA 1-19: Selector Matches izveštaj.

IE8 sadri ugrađene razvojne alate, tako da nije neophodno da ih naknadno instalirate. Celokupna dokumentacija je mrežno dostupna, a u trenutku dok ovo pišemo, adresa na kojoj se nalazi je samo malo jednostavnija za dešifrovanje od adrese za prethodno opisanu paletu sa alatima. Zbog toga, jednostavno ukucajte ključne reči **Discovering Internet Explorer Developer Tools** u vaš omiljeni pretraživač, i među prvim rezultatima pretraživanja prikazaće se željena adresa.

Da biste pokrenuli alat, neophodno je da selektujete Developer Tools stavku Tools menija IE čitača (prikazana je na slici 1-20), ili da pritisnete F12 taster na tastaturi. Da, to je sve što treba da uradite. Ono što se prikazuje veoma je slično IE Developer paleti sa alatima, koja je neka vrsta hibrida Firebug i Web Developer palette sa alatima. Meniji su skoro isti kao u IE7 paleti sa alatima, ali je sadržaj kartica mnogo sličniji Firebug alatu, nego IE7 paleti sa alatima.



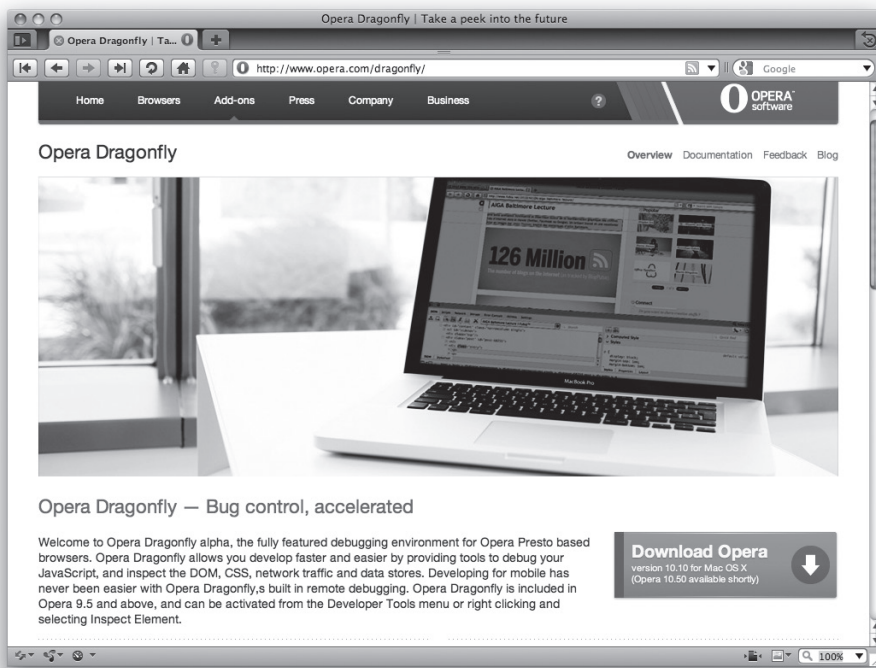
SLIKA 1-20: Razvojni alati u programu IE8.

Kada je reč o Style kartici Developer alata, može delovati zbunjujuće da su stilovi uređeni u redosledu koji nema određenog smisla. Svakako nisu uređeni na osnovu specifičnosti stilova. Listing prikazuje deklaracije koje su predefinisane drugim deklaracijama, što je lepo, ali bez smislenog uređivanja kao u Firebug alatu, veoma je teško snaći se. Ipak, iako paleta sa alatima i alati ne obuhvataju sve ono što postoji u Firebug/Web Developer paleti sa alatima, veoma su korisni i treba da budu sastavni deo instalacije svakog programera koji razvija web stranice. Ovi alati su veoma korisni prilikom praćenja izvornog koda za elemente na stranici, kao i ostalih specifičnosti Explorer web čitača.

DRAGONFLY (ZA WEB ČITAČ OPERA)

Ukoliko je vaš razvoj primarno usmeren ka Opera web čitaču, tada ćete svakako koristiti

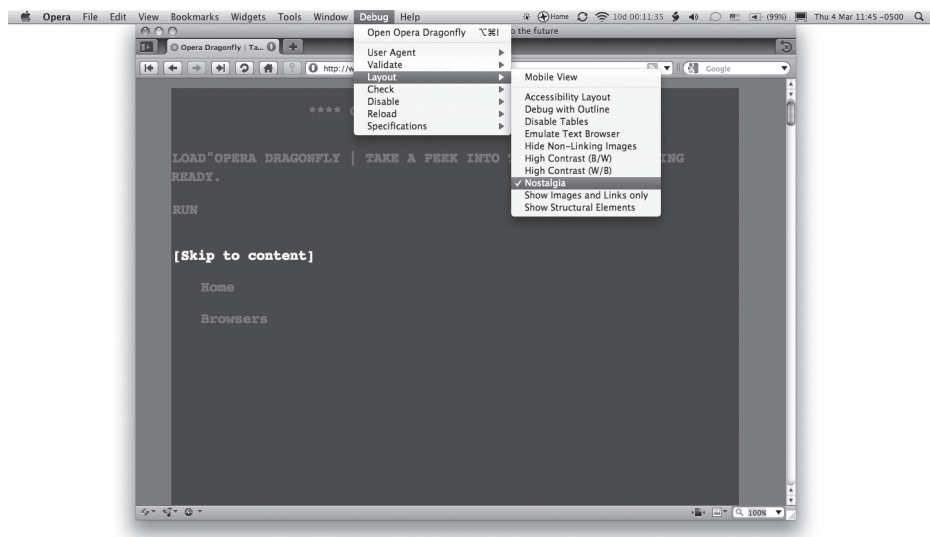
Dragonfly (slika 1-21), razvojno okruženje koje je ugrađeno u Opera 9.5 i novije verzije. Više informacija o tome možete pronaći na adresi na stranici opera.com/dragonfly.



SLIKA 1-21: Dragonfly stranica.

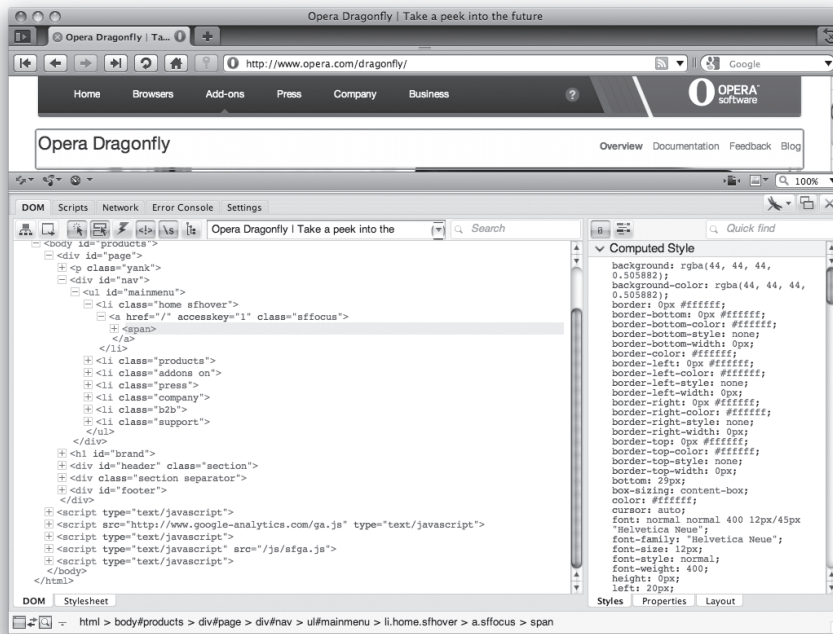
Da biste pokrenuli Dragonfly, neophodno je da pristupite Tools meniju, a zatim u Advanced opcijama selektujete Developer Tools. Međutim, možete da instalirate Debug meni tako što pristupite web stranici na adresi opera.com/dragonfly i pronađete link za instaliranje. Nakon završetka instalacije, pristupate Dragonfly alatu i ostalim funkcionalnostima pomoću prethodno navedenog menija. Možete da pritisnete Option+Command+I kombinaciju tastera (Mac korisnici) ili Option+Control+I (korisnici operativnog sistema Windows) da biste startovali alat. Postoji jedna specifičnost vezana za prečice na tastaturi: ukoliko je Dragonfly već prethodno otvoren, za njegovo zatvaranje ne možete koristiti tastaturu. Da biste izvršili zatvaranje, neophodno je da koristite miša, ili Command+W (Control+W) kombinaciju tastera. To funkcioniše onda kada je Dragonfly otvoren u posebnom prozoru. Ukoliko je Dragonfly postavljen kao deo prozora web pregledača, tada Command+W samo zatvara Dragonfly ukoliko je u fokusu, što je postignuto prethodnim selektovanjem neke funkcije alata. U suprotnom, ova kombinacija tastera dovodi do zatvaranja prozora aplikacije.

Dobra karakteristika Debug menija je da ima linkove direktno na HTML, CSS i ostale specifikacije. Još jedan zanimljiv deo je Layout podmeni, koji omogućava da Opera čitač postavite u režim rasporeda elemenata kao što je Emulate Text Browser i Show Structural Elements. Postoji čak i Nostalgia raspored elemenata (prikazan na slici 1-22), koji će zagrejati srce svih onih koji su se računarima bavili osamdesetih godina prošlog veka.

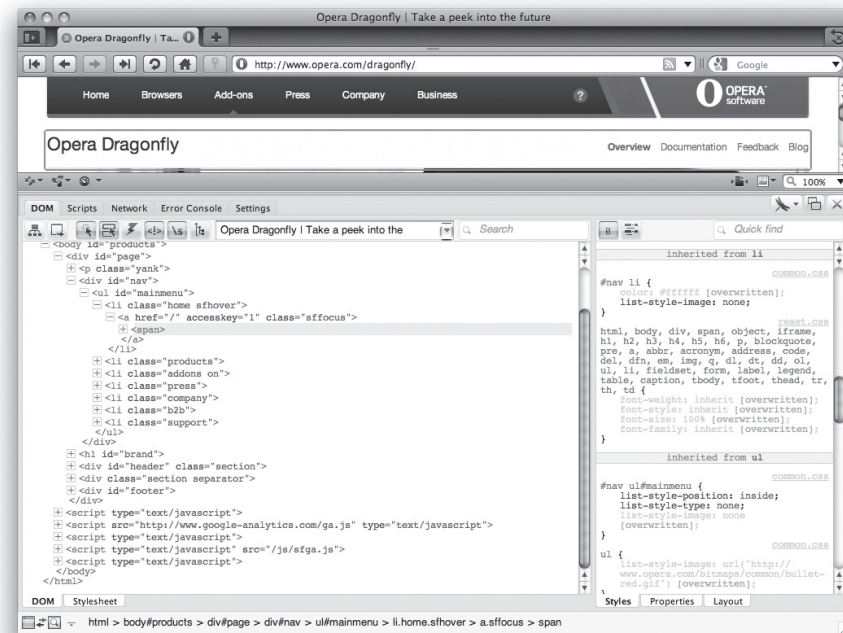


SLIKA 1-22: Dragonfly stranica u Nostalgia načinu prikazivanja.

Iako Dragonfly rasporedom elemenata mnogo podseća na Firebug, postoje određene veoma primetne razlike. Na prvom mestu, Styles kartica, koja se nalazi na desnoj strani, može da prikazuje izračunate stilove zajedno sa deklarisanim stilovima (videti sliku 1-23), a svaka grupa može da se proširi ili skupi. Kao i kod Firebug alata, sve ono što pregledate nije istovremeno i ono što je deklarisan, jer su skraćeno definisana svojstva prikazana u proširenom obliku uz prikazivanje pojedinačnih svojstava. Dobra karakteristika je da Dragonfly prikazuje skraćeno definisana svojstva ukoliko želite da ih pregledate – makar za one stilove čije se karakteristike određuju računskim putem. Jedna manje poželjna razlika u odnosu na Firebug je da su sve deklaracije koje su predefinisane drugim deklaracijama prikazane sivom bojom i žutim tekstom [overwritten] pored nje (prikazano na slici 1-24). To komplikuje stvari i značajno otežava pregledanje vrednosti u predefinisanim deklaracijama.

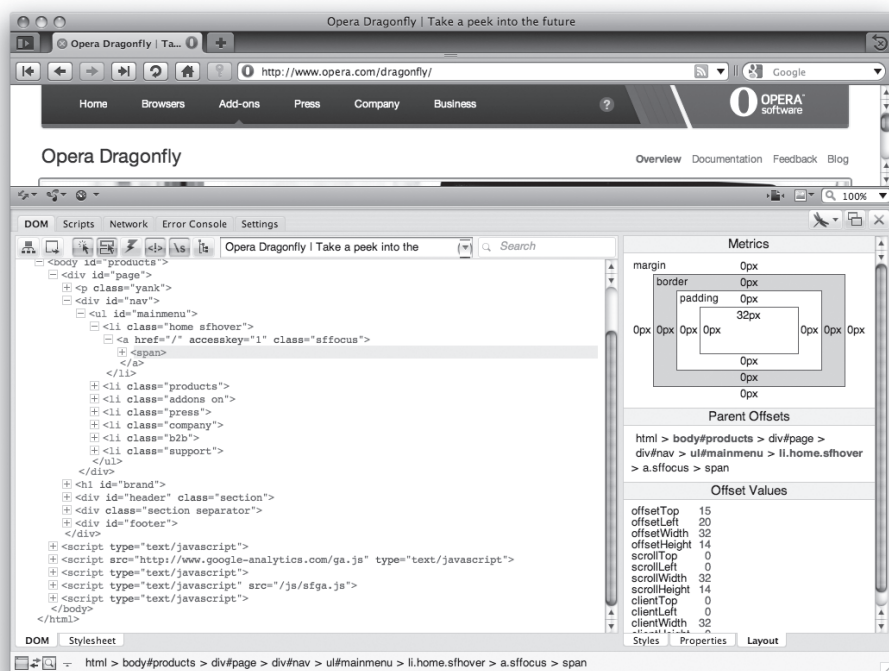


SLIKA 1-23: Dragonfly sa proširenom Computed Style grupom.



SLIKA 1-24: Predefinisani stilovi prikazani u Styles kartici.

Na slici 1-25 prikazana je Layout kartica, na kojoj je predstavljen raspored za selektovani element. Pored toga, prikazane su i dimenzije elementa, odnosno vrednosti izražene u pikselima za različita svojstva, kao što su `offsetTop` i `scrollLeft`.

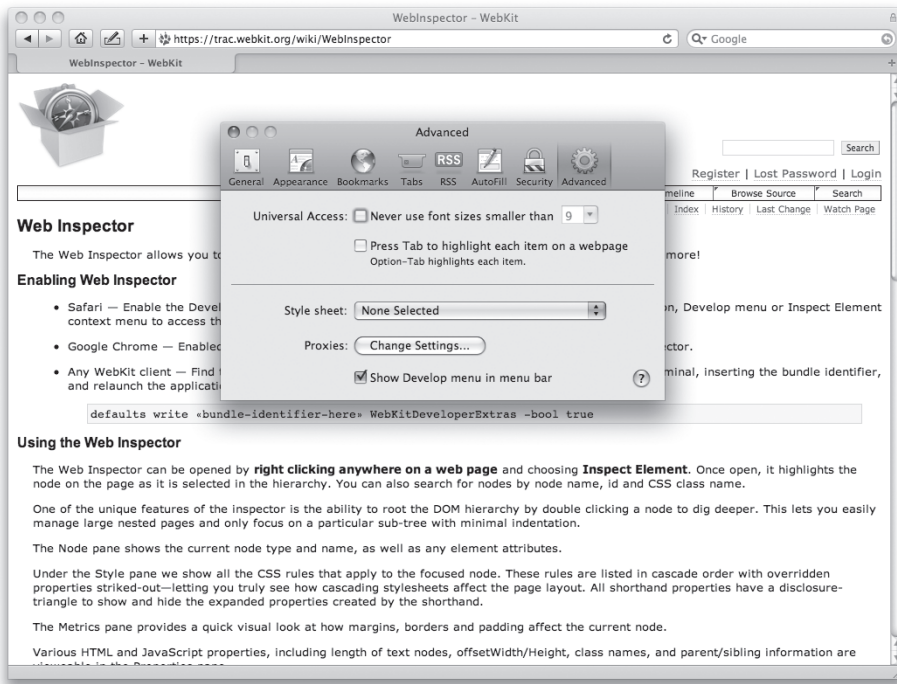


SLIKA 1-25: Sveobuhvatna Layout kartica Dragonfly alata.

WEB INSPECTOR (SAFARI ČITAČ)

Ukoliko prevashodno koristite Safari prilikom razvoja web aplikacija, tada je neophodno da u pomoć pozovete Web Inspector.

Da biste aktivirali Web Inspector, pristupite Preferences stranici Safari čitača, selektujte Advanced stavku, a zatim selektujte polje pored Show Develop menija u liniji menija (prikazano je na slici 1-26). Nakon što to uradite, možete da pokrenete Web Inspector tako što selektujete Show Web Inspector u Develop meniju, ili pritisnete Option+Command+I kombinaciju tastera. Kao i kod Dragonfly alata, korišćenje tastature je pomalo problematično: ukoliko je Web Inspector već prethodno otvoren, za njegovo zatvaranje ne možete koristiti tastaturu. Za to je neophodno da koristite miša. Command+W kombinacija tastera ne funkcioniše ukoliko se Web Inspector ne nalazi u posebnom prozoru; ukoliko je alat deo glavnog prozora web čitača, pritiskanjem prethodno pomenute kombinacije tastera zatvara se prozor aplikacije.

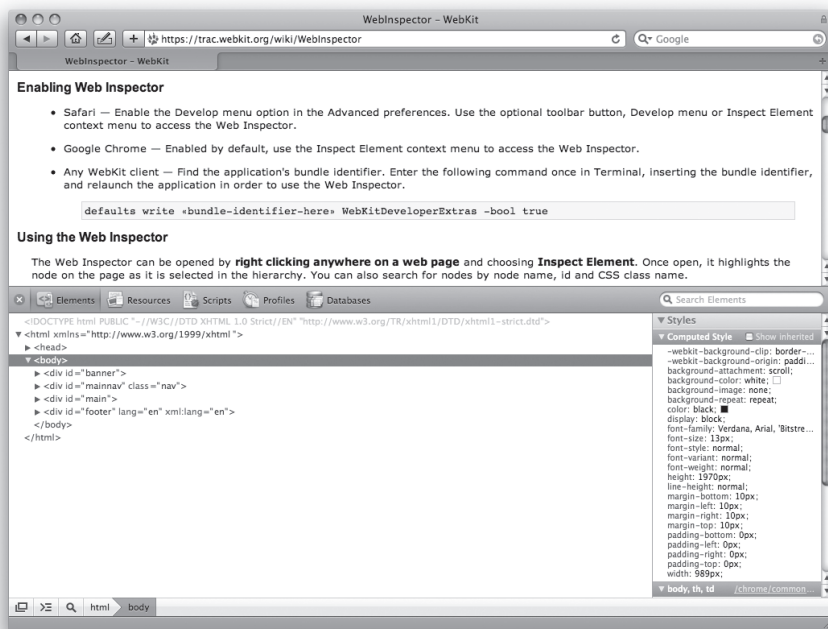


SLIKA 1-26: Omogućavanje Develop menija.

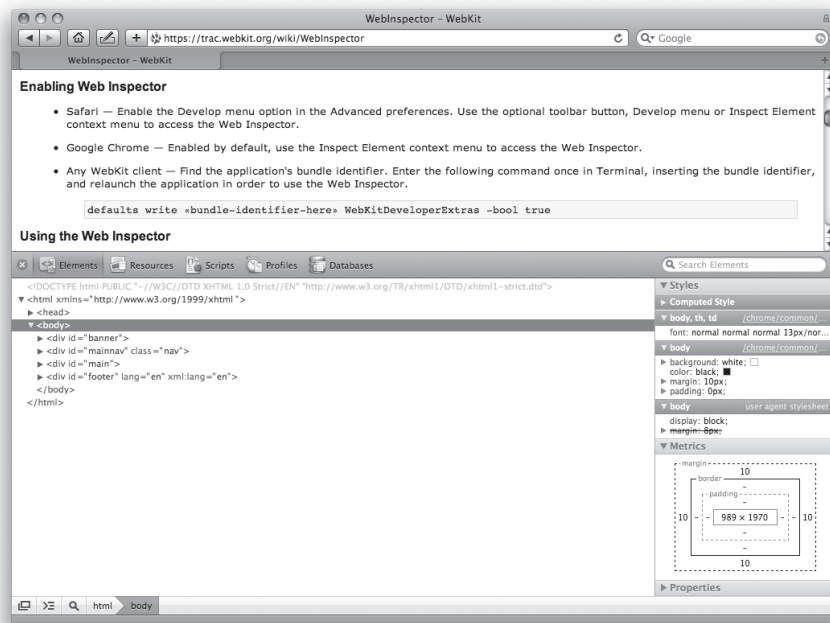
Iako raspored elemenata Web Inspector alata u potpunosti podseća na Firebug, postoje određene značajne razlike. Na primer, panel sa desne strane sadrži stilove čije su karakteristike definisane izračunavanjem, kao i grupe (videti sliku 1-27). Kao i kod Firebug alata, sve ono što se ovde prikazuje nije istovremeno i deklarirano, a skraćeno definisana svojstva su proširena tako da se prikazuju sva pojedinačna svojstva, uz istovremeno prikazivanje skraćeno definisanih svojstava. Takođe, ukoliko selektujete

Show Inherited polje, očekujte veoma dugu listu.

Neposredno ispod toga, svako pravilo koje se primenjuje na posmatrani element prikazano je kao izdvojena grupa. Možete da proširite ili skratite prikaz svake grupe. Ispod toga nalazi se Metrics podpanel, koji prikazuje dimenzije elementa koji se pregleda (videti sliku 1-28).



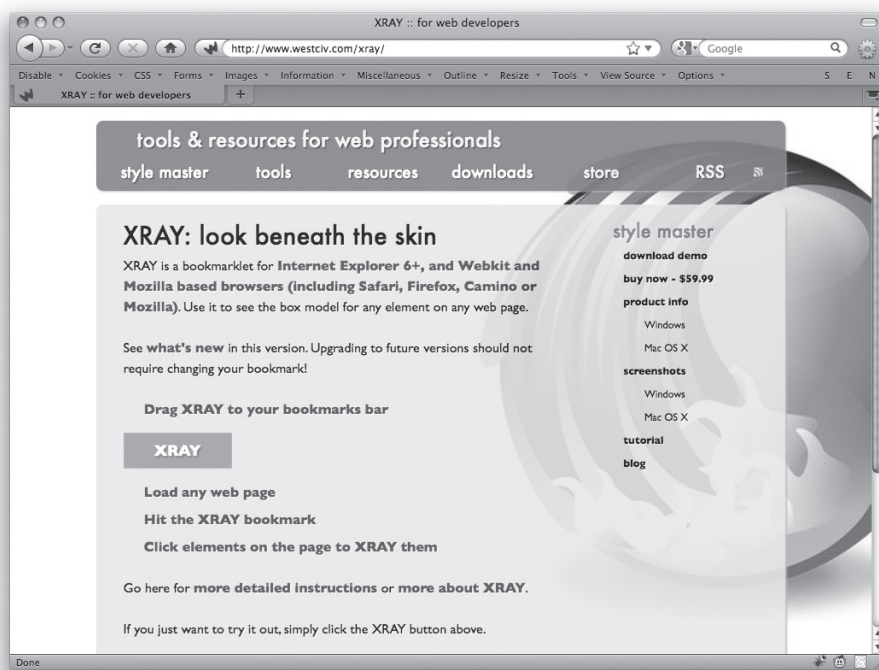
SLIKA 1-27: Stilovi čije se karakteristike određuju.



SLIKA 1-28: Regularni stilovi i grupe elemenata.

XRAY ALAT

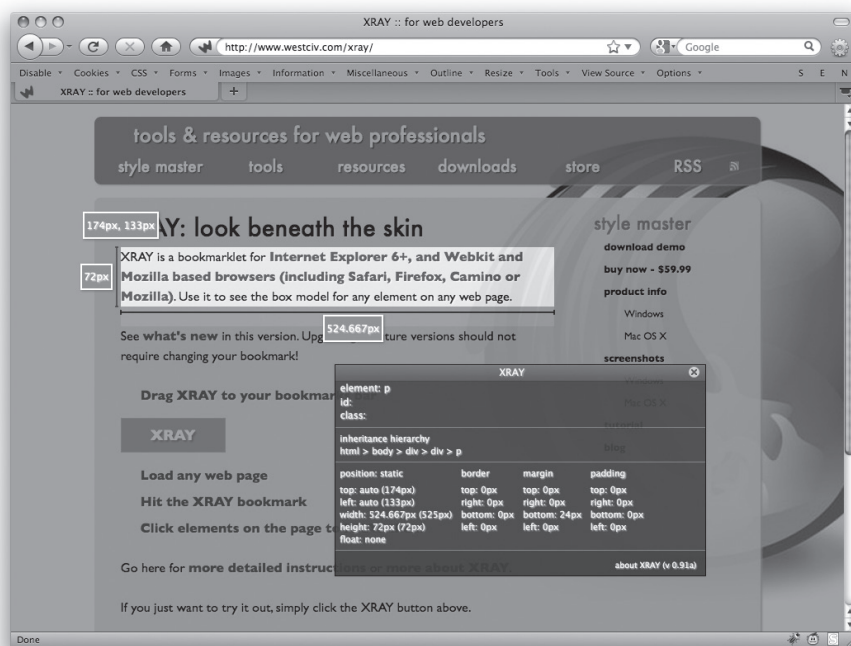
Ukoliko ste u potrazi za jednostavnim alatom za praćenje elemenata, koji se može primenjivati u više web čitača, tada je XRAY, koji je prikazan na slici 1-28, idealno rešenje. Ima veoma ograničeno polje primene, ali je to ujedno i njegova glavna prednost, pod uslovom da su vam neophodne baš one funkcionalnosti koje ovaj alat pruža.



SLIKA 1-29: XRAY stranica.

Pristupite web stranici koja se nalazi na adresi [westciv.com/xray](http://www.westciv.com/xray), a zatim prevucite veliko XRAY polje na vašu bookmarks paletu sa alatka (ili meni, ukoliko želite da ga sakrijete). Nakon toga, svaki put kada pregledate određenu stranicu i želite da analizirate određeni element, pokrenite XRAY alat, a zatim selektujte element koji vas interesuje.

Nakon što ste selektovali određeni element, on postaje posebno naglašen, a sa strane se prikazuju informacije o njegovim dimenzijama, kao i XRAY polje, koje je vidljivo na slici 1-30, i koje prikazuje određene dodatne informacije vezane za položaj u stablu dokumenta, bilo koju vrednost identifikatora ili klase, odnosno osnovni skup vrednosti CSS svojstava. Ukoliko selektujete bilo koji od elemenata u hijerarhiji nasleđivanja, XRAY omogućava pristup željenom elementu. Da biste se privremeno otarasili XRAY alata, do sledećeg puta kada vam je neophodan, jednostavno kliknite close ikonicu koja se nalazi u gornjem desnom uglu polja.



SLIKA 1-30: XRAY alat u akciji.

Sličan alat, ali sa drugom primenom, je MRI (westciv.com/mri), koji vam omogućava da unesete selektor, nakon čega se prikazuju svi elementi koje selektor identifikuje.

SELECTORACLE ALAT

Naziv SelectORacle (videti sliku 1-31) zvuči kao dobra reklama za određenu bazu podataka, ali nije: umesto toga, radi se o mrežno dostupnom alatu koji prevodi validne selektore u nešto što podseća na standardni engleski jezik. (Naziv alata nastao je spajanjem engleskih reči “Selector” i “Oracle”.)

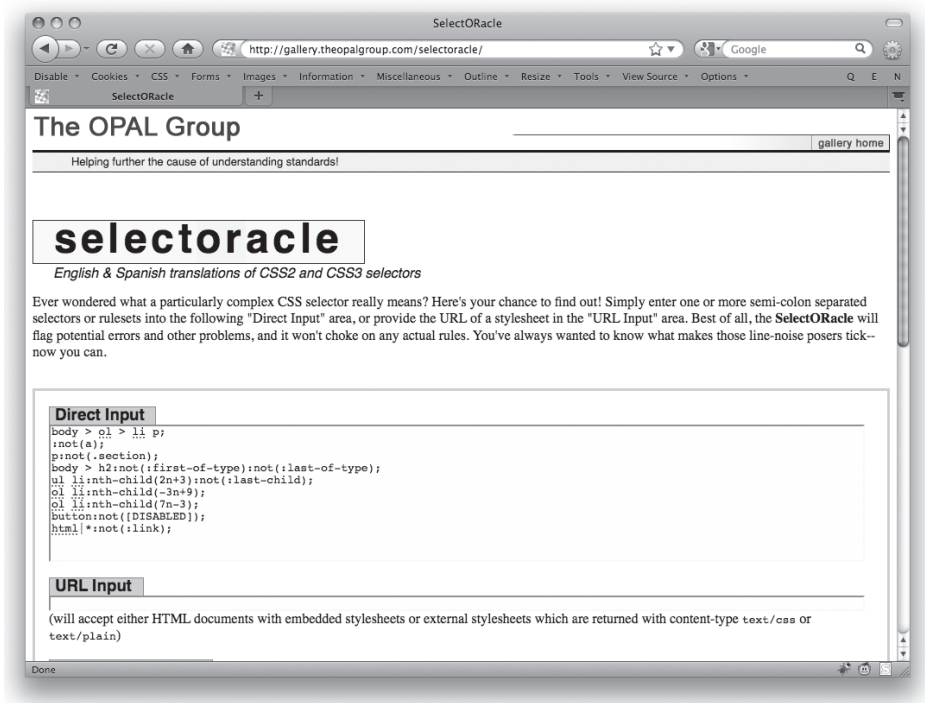
Pristupite web stranici koja se nalazi na adresi gallery.theopalgroup.com/selecto-acle i unesite jedan ili više validnih CSS selektora, nezavisno od toga koliko su oni složeni. Koristite engleski jezik ili izaberite španski jezik, a zatim kliknite Explain This! Taster. Kao rezultat dobijate objašnjenje vezano za svaki uneti selektor. Na primer, za:

```
ul li:nth-child(2n+3):not(:last-child)
```

... kao rezultat izvršavanja dobija se sledeće objašnjenje:

Selects any `li` element that is an odd numbered child starting with the third child and that is not a last child that is a descendant of a `ul` element.

(Selektuje bilo koji `li` element, koji je nižeg nivoa, označen neparnim brojem koji počinje od trećeg izvedenog elementa, i koji nije poslednji izvedeni element koji sledi `ul` element.)



SLIKA 1-31: SelectORacle stranica.

U redu, ovo možda deluje pomalo zbunjujuće na prvi pogled, ali ukoliko detaljno pročitate tekst, (uglavnom) će vam biti jasno o čemu se radi. Takođe, u trenutku dok ovo pišemo, nijedan postojeći web čitač nije u stanju da podrži sve ono što omogućava selektor, tako da nemojte da se previše brinete zbog toga. Treba da znate da ukoliko postoji mogućnost da vam namena određenog selektora nije potpuno jasna, postoji mogućnost da vam SelectORacle pruži neophodna objašnjenja.

Da li ste ikada poželeli da vas web čitač obavesti prilikom učitavanja stranice koja sadrži zastarele, nespecifične elemente kao što je font? Sada je to moguće, i tu u vizuelnom obliku, korišćenjem dijagnostičkih stilova.

DIJAGNOSTIČKI STILOVI

Primenom dijagnostičkog stila, možete imati brzi vizuelni uvid u probleme koji su vezani za kod vaše stranice. Jedan primer dijagnostičkog stila može se pronaći na web stranici na adresi meyerweb.com/eric/tools/css/diagnostic (u obe verzije, CSS3 i prilagođene za IE7 čitač), a sličan resurs se može pronaći na web stranici koja se nalazi na adresi accessites.org/site/2006/07/big-red-angry-text.

Šta je poenta? Na primer, jedna linija meyerweb dijagnostičkog stila je oblika:

(Strana 32 – kod)

```
*[style], font, center {outline: 5px solid red;}
```


Prethodnim kodom se postavlja deblja, puna linija crvene boje oko svakog elementa koji sadrži `style` atribut, svakog `font` elementa i `center` elementa. Sve to možete dodatno da začinite sa `background-color: lime` kako biste doprli do suštine. Ideja je da se identifikuju mesta na kojima se pojavljuje problematičan kod, bilo putem unosa u CMS sistem ili na neki drugi način.

Možda ćete pomisliti kako validacija može da vas reši problema vezanih za kod stranice, ali to uvek nije tačno. Istina, imaćete dovoljno informacija o fontovima, ali postoje brojni problemi koje validator ne može da identifikuje. Razmotrimo sledeći primer, u kome je prikazan često korišćeni JavaScript link:

```
<a href="#" onclick="javascript:nextPage();" >Next</a>
```

U prethodnom kodu validator neće identifikovati nikakve probleme, pošto je sam kod u potpunosti ispravan. Problem je što za sve korisnike koji nemaju JavaScript, link neće funkcionisati. Trebalo bi da postoji i deo koda koji se izvršava onda kada JavaScript ne daje nikakve rezultate, a to bi trebalo da se realizuje pomoću `href` vrednosti. Zbog toga, sledeći meyerweb dijagnostični stil ima oblik:

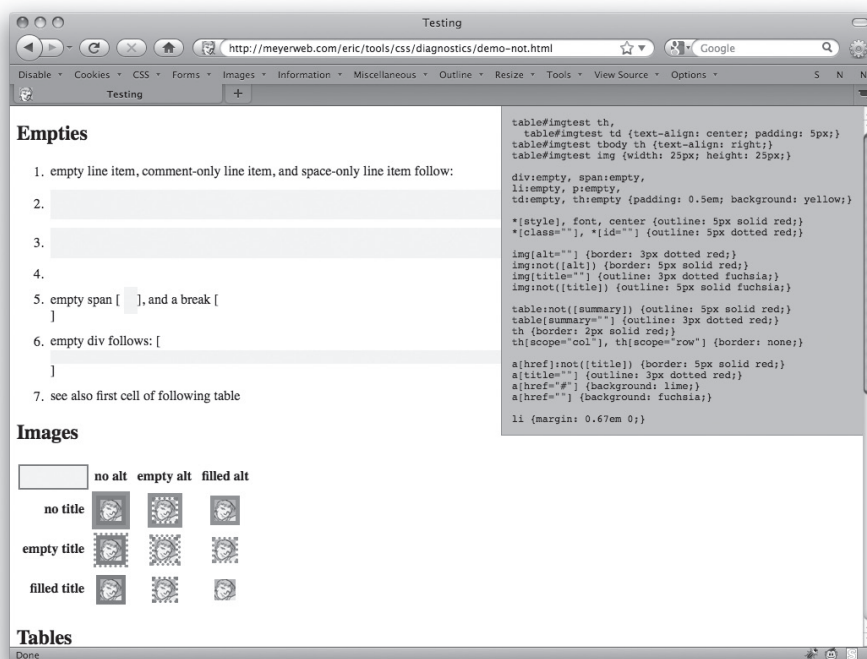
```
a[href="#"] {background: lime;}
```

Prethodni kod će identifikovati sve linkove koji ne sadrže povratnu vrednost u situaciji u kojoj se ne izvršava JavaScript kod, a zatim koristiti vrednost `href` atributa. (Funkcioniše korišćenjem selektora atributa; više informacija o tome možete pronaći u odeljku "Jednostavna selekcija atributa", u Poglavlju 2.)

Na koji način možete da dijagnostikujete CSS kod? Importovanjem u CSS stranice koju razvijate, a zatim uklanjanjem pre nego što aktivirate stranicu, ili podešavanjem korisničkog stila u vašem web čitaču, tako da možete da ga primenjujete na bilo koju stranicu kojoj budete pristupali.

U sledećem primeru prikazan je celokupan dijagnostički stil koji funkcionise tako što pronalazi elemente koji nemaju sadržaj, uklanjanje slike koje ne sadrže `alt` ili `title` attribute kao i one koje se ne koriste, pronalazi tabele bez `summary` atributa i zaglavlja tabele koja imaju pogrešne vrednosti dometa, odnosno linkove koji ne vode do odredišta ili nemaju definisane `title` ili `href` attribute. Obratite pažnju na to da ova verzija koda neće funkcionisati u IE7 čitaču, zbog postojanja selektora atributa. Ova verzija ne funkcionise ni u IE8 čitaču, zbog korišćenja `:not()` i `:empty()` pseudo-klasa. Na slici 1-32 prikazana je probna stranica za ovaj dijagnostički CSS.

```
div:empty, span:empty,
li:empty, p:empty,
td:empty, th:empty {padding: 0.5em; background: yellow;}
*[style], font, center {outline: 5px solid red;}
*[class=""], *[id=""] {outline: 5px dotted red;}
img[alt=""] {border: 3px dotted red;}
img:not([alt]) {border: 5px solid red;}
img[title=""] {outline: 3px dotted fuchsia;}
img:not([title]) {outline: 5px solid fuchsia;}
table:not([summary]) {outline: 5px solid red;}
table[summary=""] {outline: 3px dotted red;}
th {border: 2px solid red;}
th[scope="col"], th[scope="row"] {border: none;}
a[href]:not([title]) {border: 5px solid red;}
a[title=""] {outline: 3px dotted red;}
a[href="#"] {background: lime;}
a[href=""] {background: fuchsia;}
```

SLIKA 1-32: Probna stranica za CSS dijagnostikovanje.

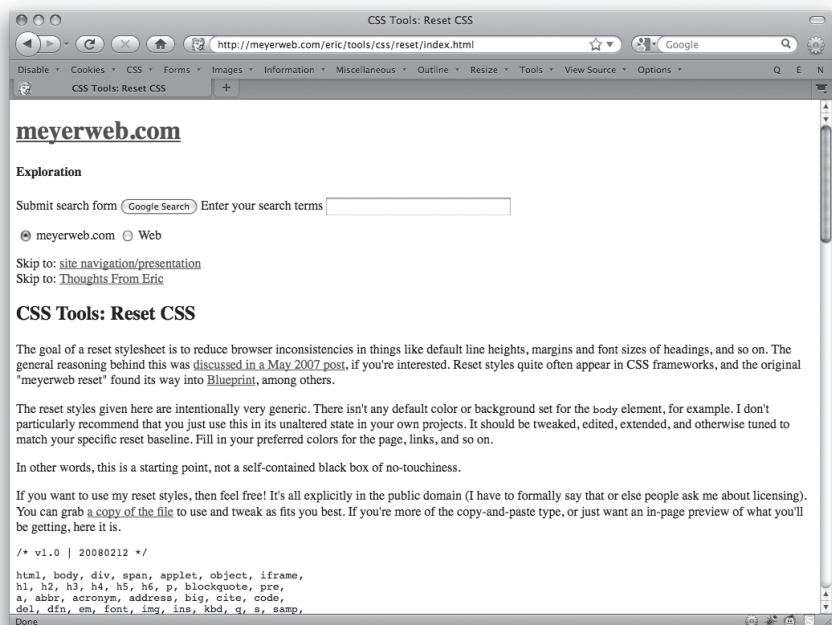
REBOOT STILOVI

Prilikom razmatranja CSS koda, do sada možda niste uzimali u obzir da se on uvek primenjuje na dokumente, čak i onda kada kreirate HTML dokument koji ne sadrži nijednu jedinu liniju CSS koda. Zapravo, postoji dosta CSS koda koji se primenjuje na dokumente bez definisanih stilova (kao što je onaj koji je prikazan na slici 1-33), a sav taj kod je definisan u okviru samog web čitača koji koristite. Podrazumevana veličina i font kojim se prikazuju zaglavlja, separacija između elemenata i linija teksta, navođenja u listama stavki, pa čak i različita vrsta polja definišu se na osnovu podrazumevanih stilova.

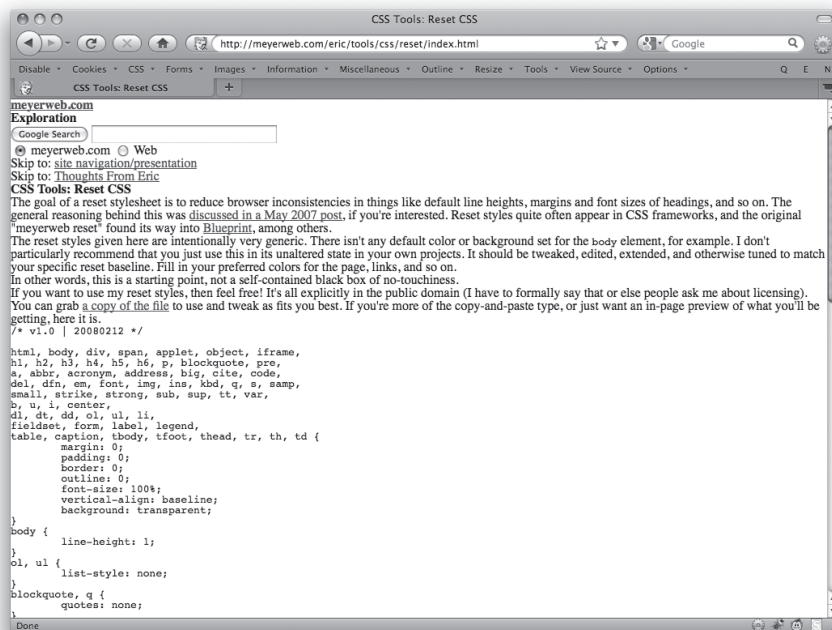
Naravno, podrazumevani stilovi se razlikuju od čitača do čitača. To nije problem vezan za web čitače, zato što ne postoji specifikacija koja bi definisala način na koji inicijalno treba da se primenjuju stilovi na dokumente. Na osnovu toga, najveći broj web čitača simulira na najbolji mogući način ono što je činio Mosaic sa dokumentima. Da, Mosaic –to je Netscape 1.0 pokušao da simulira, to je pokušao da simulira IE3, i tako redom do današnjih dana. Ukoliko detaljnije proučite podrazumevane stilove, utvrdićete da je većina stvari, zapravo, potpuna kopija onoga što je definisano u prvim beta verzijama Mosaic čitača, i to piksel po piksel.

Kao odgovor na to, određeni korisnici su razvili reset stilove (videti sliku 1-34), čiji je cilj da smanje brojne nekonzistentnosti eksplicitnim definisanjem vrednosti standardnih svojstava. Najjednostavniji primer za to je:

```
* {margin: 0; padding: 0;}
```



SLIKA 1-33: Dokument bez primenjenog stila (ali, zapravo, veoma lepo stilizovan).



SLIKA 1-34: Dokument sa primenjenim osnovnim reset CSS kodom.

Veliki broj korisnika primenjuju reset stilove, pošto je postupak primene veoma jednostavan. Problem je što se oni primenjuju na sve elemente u dokumentu, uključujući i elemente forme, kao što su polja za unos teksta i polja za selektovanje. Pošto web čitači trenutno primenjuju CSS kod na elemente forme na različite načine (a neki od čitača ne primenjuju uopšte), pristup zasnovan na „svim elementima“ iz prethodne rečenice dovodi to toga da forme postaju veoma nekonzistentne, kao rezultat pokušavanja da se web čitači učine konzistentnijim pomoću resetovanja podešavanja.

Zbog toga su razvijeni složeniji oblici resetovanja. Jedan veoma popularan reset skript raspoloživ je na adresi meyerweb.com/eric/tools/css/reset. On počinje na sledeći način:

```
html, body, div, span, applet, object, iframe,
h1, h2, h3, h4, h5, h6, p, blockquote, pre,
a, abbr, acronym, address, big, cite, code,
del, dfn, em, font, img, ins, kbd, q, s, samp,
small, strike, strong, sub, sup, tt, var,
b, u, i, center,
dl, dt, dd, ol, ul, li,
fieldset, form, label, legend,
table, caption, tbody, tfoot, thead, tr, th, td {
    margin: 0;
    padding: 0;
    border: 0;
    outline: 0;
    font-size: 100%;
    vertical-align: baseline;
}
body {
    line-height: 1;
}
```

Sjajno! To je zaista veliki broj elemenata. Razlog navođenja toliko elemenata je selektovanje svih elemenata koji nisu ulazni elementi formi (`input`, `select`, `textarea`), a zatim isto-
vetno funkcionisanje tih elemenata. Prvo pravilo eksplicitno definiše margine, pomeranje, ivice i konture tako da imaju vrednost 0, odnosno primorava sve elemente da koriste font iste veličine i isto vertikalno poravnanje teksta. Drugo pravilo definiše da `body` element ima redukovanu visinu, a da se ta vrednost nasleđivanjem prenosi na sve elemente koji su u nivoima ispod `body` elementa.

Postoje i dodatna pravila u `meyerweb reset skriptu`, a vezana su, između ostalog, za uklanjanje tačaka za navođenje i automatski generisanih navodnika oko `blockquote` i `q` elemenata. Ponovimo, cilj je da svi web čitači, koliko god je to moguće, na isti način prikazuju stranicu pre nego što pristupite ulepšavanju njenog prikaza korišćenjem CSS koda.

U ovom trenutku ćete možda pomisliti: „Samo malo, to znači da moramo da poništimo sve ono što smo prethodno definisali! Ne želim da moja stranica ima jediničnu visinu linije – to je isuviše zgusnut prikaz! Ne želim ni da se bavim vertikalnim podešavanjem položaja indeksa i stepena!“

Potpuno ste u pravu. Neophodna je promena stila, tako da ispunjava vaše zahteve. Recimo da uvek počinjete sa visinom linije vrednosti 1.4, kako bi tekst bio dovoljno pregledan. Da biste to postigli, neophodno je da izmenite samo jednu liniju skripta za resetovanje:

```
body {  
    line-height: 1.4;  
}
```

Kada već menjate skript, onda možete da dodate vašu standardnu pozadinu za stranice i boje teksta, uz primenu vašeg omiljenog fonta:

```
body {  
    font: smaller/1.4 Helvetica, sans-serif;  
    background: #ABACAB;  
    color: #444;
```

Možete da definišete pravilo kojim se uklanjaju oznake ispred stavki u listama navođenja, jer smatrate da nikada više nećete menjati ovo podešavanje. (To ne važi za mene, ali svi smo različiti, zar ne?) Dodajte pravila koja definišu način prikazivanja stavki u listi, rastojanje između paragrafa i stavki u listama, način na koji želite da prikazujete podatke, i tako redom.

Ono što ste do sada definisali nije reset stil, već reboot stil. Vi, zapravo, definišete nova inicijalna podešavanja vezana za definisanje stilova dokumenta, uspostavljajući tako novu početnu tačku za razvoj svih ostalih projekata. Nakon što definišete ovaj stil za definisanje novih inicijalnih podešavanja, možete da pristupite razvoju novog projekta, tako da stil postane neka vrsta jezgra oko koga će se razvijati svi novi stilovi.

Ne samo da možete da definišete nova inicijalna podešavanja koja vaš web čitač koristi prilikom primene CSS dokumenata, već možete i da nadgradite neka od njih koristeći JavaScript. To je zaista moguće.

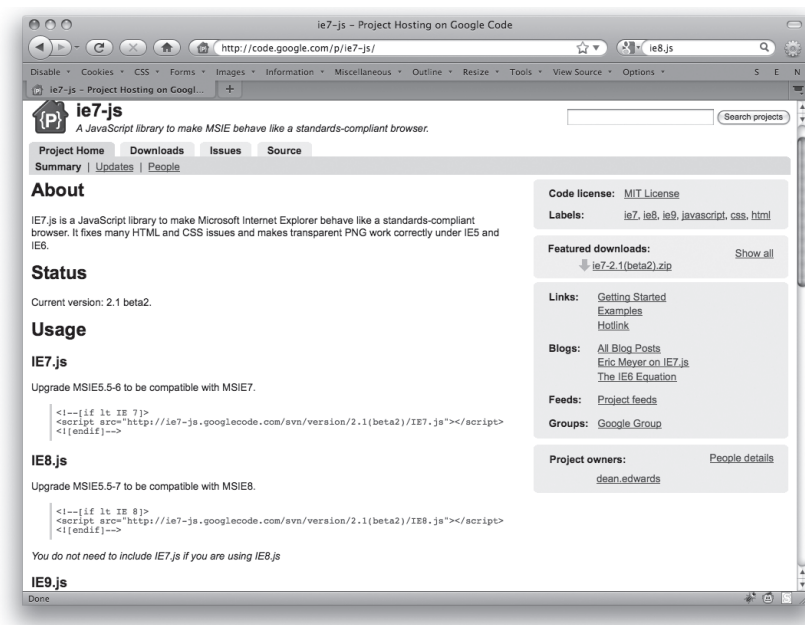
IE9.JS SKRIPT

Ukoliko koristite IE9.js skript, čiji je autor Din Edvards (Dean Edwards), možete da podesite web čitače IE5 do IE8 tako da funkcionišu slično kao IE9 (koji, u trenutku dok ovo pišemo, i dalje nije izašao) kada se radi o rukovanju CSS i HTML kodom. Ovaj skript možete pronaći na adresi code.google.com/p/ie7-js/; da, ie7 deo u nazivu nije greška (videti sliku 1-35). (Naziv je takav zato što je projekat započet kao IE7.js, a nakon što su se pojavili IE8 i IE9 čitači, bilo je neophodno kreirati nove verzije skripta.)

IE9.js je skup JavaScript rutina koje, ukoliko je verzija Internet Explorer web čitača starija od IE9, skeniraju CSS i HTML kod stranice i utvrđuju koji delovi koda nisu podržani u verziji IE čitača koji se koristi za pregledanje sadržaja stranice. Postoji veliki broj različitih podešavanja koja omogućavaju da se ova podrška obavlja transparentno.

Na primer, IE5 i IE6 čitači ne podržavaju selektore atributa. Zbog toga, ukoliko imate pravilo sledećeg oblika:

```
a[href] {text-decoration: none; color: red;}
```



SLIKA 1-35: IE7.js stranica.

...IE5 i IE6 čitači će ignorisati u potpunosti selektore atributa, a vaši linkovi će ostati neproimenjeni. (Iako bi ovo usrećilo Jakoba Nilsena, dizajner projekta bi bio manje impresioniran.)

Ukoliko koristite IE9.js, ovaj privlačni skript obavlja podešavanja tako da IE5 i IE6 mogu da primenjuju stilove na linkove, tako da to jednostavno funkcioniše. Neophodno je samo povezivanje sa IE9.js skriptom iz svake stranice za koju je to neophodno, i sve će biti u redu.

Naravno, ništa od toga neće imati efekta ukoliko nije omogućeno korišćenje JavaScript koda, što znači da je neophodno da izvagate prednosti korišćenja skripta u odnosu na ono što korisnici očekuju od vaše strance. Da li veliki broj vaših korisnika koristi IE6 web čitač? Da li oni vole da im u web čitaču bude onemogućeno korišćenje JavaScript koda? I tako redom. Naravno, to je isto što je neophodno razmatrati i prilikom samog projektovanja web stranice, tako da ste makar upoznati sa samim procesom odlučivanja.

Standardno se preporučuje korišćenje script elementa koji omogućava pristupanje JavaScript datoteci, i to u okviru uslovne naredne, kao što je prikazano u sledećem primeru:

```
<!--[if lt IE 9]>
<script src="/code/IE9.js" type="text/javascript"></script>
<![endif]-->
```

Skript će se izvršavati samo kada je to neophodno, tako da možete da preskočite uslovne komentare. Ukoliko postupite na prethodno opisani način, to znači da će svaki korisnik preuzeti skript, nezavisno od toga što njegov ili njen čitač možda nikada neće izvršiti taj skript. Primenom uslovnih komentara obezbeđujete da samo oni čitači koji mogu da izvršavaju skript pristupaju njegovom preuzimanju.

Kao što je napomenuto, postoje i prethodne verzije skripta, koje možete da primenjujete kako bi vaš IE čitač bio na nivou funkcionalnosti koje pružaju IE7 ili IE8 verzije čitača. Ukoliko utvrdite da IE9.js skript ne zadovoljava vaše potrebe, isprobajte neku od prethodnih verzija.