

# UPOZNAJTE SE SA JAVASCRIPTOM

---

# 1

Za kreatore veb lokacija, razvoj HTML-a na neki način je predstavljao blagoslov. U prvim danima World Wide Weba, HTML je bio prilično jednostavan i bilo je lako naučiti gotovo sve što je potrebno za sklapanje veb stranica.

Kako se veb razvijao, tako su rasle i aspiracije veb dizajnera, a njihovi zahtevi za većom kontrolom nad izgledom stranice usloveli su da se HTML menja i postane složeniji.

Budući da je veb dinamičan medijum, dizajneri stranica su želeli da njihove stranice takođe budu i interaktivne za korisnike, tako da je uskoro postalo očigledno da HTML nije dovoljan da se ispune ovi zahtevi. Netscape je uveo JavaScript kao način za kontrolu pretraživača i veb stranicama dodao dinamičnost i interaktivnost.

Od svog nastanka, JavaScript je prilično evoluirao (mada povremeno u različitim pravcima, sve u zavisnosti od pretraživača). Razvoj JavaScripta detaljnije ćemo razmatrati kasnije u ovoj knjizi. Za sada je dovoljno reći da je JavaScript tokom godina postajao sve moćniji, i da je zauzeo svoje mesto među alatima svakog ozbiljnog veb autora.

U ovom poglavlju naučićete šta je JavaScript (i šta nije); šta on može da uradi (a šta ne može); kao i neke osnove jezika JavaScript.

# Šta je JavaScript

JavaScript predstavlja programski jezik koji možete koristiti za dodavanje interaktivnosti svojim veb stranicama. Ali ukoliko niste programer, ne paničite zbog izraza „programski jezik”; postoji puno JavaScriptova koji su dostupni na Webu, a koje možete da kopirate, i uz minimalan napor modifikujete za sopstvene potrebe. U suštini, ovaj način „zavirivanja preko ramena” drugih programera, predstavlja odličnu tehniku za upoznavanje sa JavaScriptom, a ako ste u tome uporni (pa pročitate ovu knjigu), uskoro ćete praviti sopstvene skriptove.

Kako bi vam olakšali početak rada sa JavaScriptom, autori su uspostavili veb lokaciju koja dopunjuje ovu knjigu. Tu smo uključili sve skriptove iz ove knjige (tako da ne morate sami da ih prekucavate!), kao i dodatne napomene, dodatke i ažuriranja. Našu lokaciju možete pronaći na adresi <http://www.javascriptworld.com>.

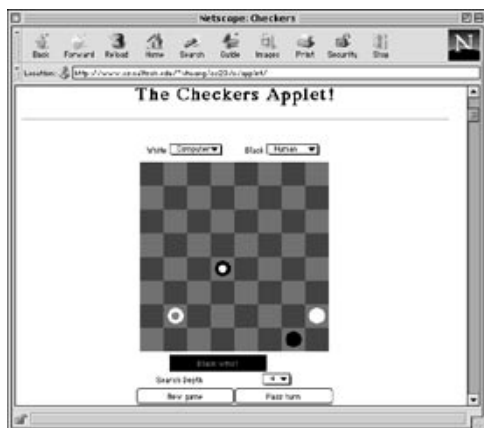
Često ćete videti da se o JavaScriptu govori kao o „skript jeziku”, pri čemu se podrazumeva da je na neki način jednostavnije pisati skriptove nego programirati. U ovom slučaju to predstavlja razdvajanje bez prave razlike. JavaScript skript je program koji je uključen u neku HTML stranu. Pošto je uokviren oznakom `<script>`, tekst tog skripta se ne pojavljuje na ekranu korisnika, a veb pretraživač zna da izvršava JavaScript program. Oznaka `<script>` najčešće se nalazi u odeljku `<head>` HTML stranice, mada možete, ukoliko želite, da postavite skriptove i u odeljak `<body>`. Skriptove koji ispisuju tekst na ekranu ili pišu HTML, najbolje je smestiti u odeljak `<body>` kao u **skriptu 1.1**. Ukoliko vam nisu poznati ovi koncepti HTML-a i ako vam je potrebno više informacija o HTML-u, predlažemo vam da pogledate knjigu Elizabeth Castro, *HTML for the World Wide veb, Fifth Edition, with XHTML and CSS: Visual QuickStart Guide*, koju je objavio Peachpit Press.

**Script 1.1** Ovaj veoma jednostavan skript ispisuje „Hello, Cleveland!” u prozoru pretraživača.

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
→ Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title>Barely a script at all</title>
</head>
<body bgcolor="#FFFFFF">
<h1>
    <script language="Javascript"
    → type="text/javascript">
        document.write("Hello, Cleveland!")
    </script>
</h1>
</body>
</html>

```



**Slika 1.1** Ovaj Java aplet igra uobičajenu partiju dame, a to je nešto što ne možete da napravite sa JavaScriptom. Aplet zauzima prostor počev od pomoćnih menija za boju igrača, do dugmadi na dnu ekrana.

## JavaScript nije Java

Uprkos imenu, JavaScript i Java nemaju skoro ništa zajedničko. Java je potpuni programski jezik, koji je za tržište razvila kompanija Sun Microsystems. Pomoću Jave, potomka programskih jezika C i C++, programeri mogu da naprave čitavu aplikaciju i kontrolišu elektronske uređaje potrošača. Za razliku od drugih jezika, Java se drži obećanja vezanih za rad na različitim platformama. Drugim rečima, programer bi trebalo da bude u stanju da napiše jedan Java program koji može da se izvršava na bilo kojoj vrsti mašine, bez obzira na to da li se na toj mašini izvršava Windows, Mac OS, ili Unix. U praksi, Java nije u potpunosti ostvarila taj san, uglavnom zbog nesklada između kompanija Sun i Microsoft oko pravca razvoja ovog jezika. Prvo se Microsoft umešao jer je hteo da integriše Javu u Windows na svoj način (način koji bi po tvrdjenju kompanije Sun učinio da Java radi na jedan način u Windowsu, a na drugi način na drugim mašinama, i tako poništio osnovnu namenu Jave). Zatim je Microsoft potpuno izbacio Javu iz Windowsa, posle stvaranja svog jezika C# sličnog Javi. Dok je ova knjiga bila u štampi, sud SAD je presudio da je Microsoft neovlašćeno isključio Javu iz Windowsa i naredio da se Java vrati.

Glavna namena Jave na *strani klijenta*, odnosno u pretraživaču korisnika, jeste da napravi aplete, male programe koji se učitavaju preko Interneta i izvršavaju unutar veb pretraživača. Zbog međuplatformske prirode jezika Java, trebalo bi da se ti apleti identično izvršavaju u svakom pretraživaču koji je omogućen za Javu.

Java aplete ugrađujete u svoje veb stranice koristeći HTML oznaku `<applet>`. Kada pretraživač vidi oznaku `<applet>`, on sa servera učitava Java aplet, a taj aplet se onda izvršava u oblasti ekrana navedenoj u oznaci (**slika 1.1**).

Da biste naučili više o Javi, preporučujemo izdanje *Java 2 for the World Wide Web: Visual QuickStart Guide* od Dori Smith.

## Odakle potiče JavaScript

Ukoliko JavaScript nije povezan sa Javom, zašto onda imaju tako slične nazive? To je još jedan primer najozbiljnijih pretnji u industriji računara: pobede marketinga nad suštinom.

Kada je Netscape dodao neke osnovne sposobnosti skriptova svom veb pretraživaču Navigator, prvobitno je taj skript jezik nazvao LiveScript. Nekako u isto vreme Java je dobijala dosta publiciteta kao „sledeća velika stvar u računarstvu”. Kada je Netscape preradio pretraživač Navigator u Navigator 2 koji je izvršavao Java aplete, on je izmenio i naziv LiveScript u JavaScript, nadajući se da će preuzeti nešto sjaja vezanog za Javu. Prosta činjenica da su JavaScript i Java veoma različiti programski jezici nije sprečila marketinške genije kompanije Netscape, tako da od tada autori, kao što smo mi, zarađuju dobre pare objašnjavajući da su JavaScript i Java veoma različite stvari. Kada pomislimo na to, trebalo bi da budemo zahvalni tim ljudima.

Kada je Microsoft uvideo da je JavaScript postao veoma popularan među veb programerima, shvatio je da bi trebalo da doda neke mogućnosti skript jezika u Internet Explorer. Oni su mogli i da usvoje JavaScript, ali kao što je to veoma često slučaj, Microsoft je umesto toga napravio sopstveni jezik koji u velikoj meri radi kao JavaScript, ali nije bio potpuno isti. Ova Microsoftova verzija nazvana je JScript.



**Slika 1.2** Reakcija na prelazak mišem predstavlja sliku koja se menja kada se pokazivač miša postavi iznad nje.

U Netscape Navigatoru 3.0 i kasnijim verzijama (sve platforme) i Internet Exploreru 5.0 i kasnijim verzijama (samo Macintosh), možete da koristite JavaScript za proveru pristupa dopunskih modula pretraživača, i da usmerite korisnike na drugu stranicu, ukoliko nemaju dopunski modul koji je potreban za pregledanje vaše stranice. (Dopunske module ne možete da detektujete pomoću JavaScripta u Internet Exploreru za Windows; da biste o ovome saznali više pročitajte poglavlje 3.)

## Šta JavaScript može da radi

Postoji mnogo toga što možete da uradite pomoću JavaScripta, kako biste poboljšali interaktivnost svojih veb stranica i ostvarili bolji i mnogo uzbudljiviji utisak za korisnike vaše lokacije. JavaScript vam dopušta da napravite aktivni korisnički interfejs, dajući korisnicima povratne informacije dok se kreću kroz stranice. Na primer, verovatno ste videli lokacije sa dugmadima koja se ističu dok pomerate pokazivač miša preko njih. To se postiže preko JavaScripta, pomoću tehnike koja se naziva *reakcija na prelazak mišem* (engl. *roll-over*) (**slika 1.2**).

JavaScript možete koristiti kako biste obezbedili da korisnici unose ispravne informacije u obrasce, što može uštedeti vreme i novac pri poslovanju. Ukoliko vaši obrasci zahtevaju izračunavanja, možete ih izvršavati u JavaScriptu na mašini korisnika, bez bilo kakve obrade na strani servera. Ovo je razlika koju treba da uočite: programi koji se izvršavaju na mašini korisnika nazivaju se programi na *strani-klijenta*; programi koji se izvršavaju na serveru (uključujući takozvane CGI; više o njima kasnije) nazivaju se programi na *strani-servera*.

Sa JavaScriptom, imate mogućnost da u letu pravite namenske HTML stranice, u zavisnosti od postupaka korisnika. Recimo, da održavate lokaciju za putovanja i da korisnik, kao destinaciju, mišem izabere Havaii. Najnoviji detalji o putovanjima na Havaje treba da se pojave u novom prozoru. JavaScript kontroliše pretraživač, tako da možete da otvorite nove prozore, prikažete okvire sa upozorenjima i da smestite posebne poruke u statusnu liniju prozora pretraživača. Pošto JavaScript ima skup vremenskih i datumskih funkcija, možete da generišete satove, kalendare i dokumente sa vremenskom oznakom.

## Šta JavaScript ne može da radi

JavaScript predstavlja jezik *na strani-klijenta*; to jest, projektovan je da svoj deo posla obavlja na vašoj mašini, a ne na serveru. Stoga, JavaScript ima neka ugrađena ograničenja, pretežno iz bezbednosnih razloga:

- JavaScript ne dopušta čitanje i pisanje datoteka na mašini klijenta. To je dobro, jer svakako ne biste želeli da veb stranica bude u stanju da čita datoteke sa vašeg diska, ili da upisuje viruse na vaš disk, ili da rukuje datotekama na vašem računaru. Jedini izuzetak jeste da JavaScript može da upiše podatke u datoteku kolačića pretraživača, a čak i tada postoje ograničenja (da biste saznali više o kolačićima pročitajte poglavlje 12).
- JavaScript ne dopušta čitanje ili pisanje datoteka na mašinama servera. Postoji veliki broj slučajeva u kojim bi ovo bilo zgodno (kao što je čuvanje broja ostvarenih poseta te strane ili podataka za popunu obrasca), ali JavaScriptu nije dopušteno da to radi. Umesto toga, potrebno je da na serveru imate program koji će rukovati ovim podacima i čuvati ih. To može da bude, na primer, neki CGI napisan u jeziku kao što je Perl ili PHP, ili neki Java program.
- JavaScript ne može da zatvori prozor koji nije sam otvorio. Tako se izbegava situacija u kojoj neka lokacija preuzima vaš pretraživač, zatvarajući prozore svih drugih lokacija.
- JavaScript ne može da čita informacije sa neke otvorene veb stranice koja potiče sa drugog servera. Drugim rečima, veb stranica ne može da čita bilo koju informaciju iz otvorenih prozora i otkrije namere korisnika koji posećuju njihovu lokaciju.



**Slika 1.3** Objekat mačka (ova se zove Pixel).



**Slika 1.4** Ovaj pomoćni meni predstavlja objekat pretraživača, kojim može da se rukuje preko JavaScripta.

## Jezik sklapanja

Evo još jedne rečenice koja ništa ne znači, a koju bi trebalo razjasniti: JavaScript je *objektno-orijentisani* jezik. Pa šta to znači?

### Objekti

Kao prvo, razmotrimo objekte. *Objekat* je zapravo neka „stvar”. Mačka, računar i bicikl su objekti (**slika 1.3**) u fizičkom svetu. Za JavaScript postoje objekti kojima on rukuje u okviru veb pretraživača, kao što su prozori, obrasci i elementi određenog obrasca, kao što su dugmad i polja za potvrđivanje (**slika 1.4**).

Pošto možete da imate više mačaka, ili više prozora, ima smisla da im se dodele imena. Mada svojim ljubimcima možete da se obraćate kao Mačka #1 i Mačka #2, to je nije dobra ideja iz dva razloga: prvo, mačke je lakše razlikovati ako imaju jedinstvena imena, a drugo, to je jednostavno nepristojno. Na isti način, u svim primerima u ovoj knjizi objekti će imati svoje jedinstvene nazive.

### ✓ Savet

- Imajte na umu da se skriptovi, koje možete naći na Internetu, pozivaju na objekte kao `window[0]` i `form[1]`. Ovo je, iz već navedenih razloga, siromašan stil i otkrićete da je praćenje objekata u skriptovima mnogo jednostavnije ako im umesto brojeva dodelite imena.

### Svojstva

Objekti imaju *svojstva*. Mačka ima krzno, računar tastaturu, a bicikl točkove. U svetu JavaScripta, prozor ima naslov, a obrazac može da ima polje za potvrđivanje.

Izmena svojstva nekog objekta može da modifikuje taj objekat, a isto svojstvo može da se primeni na potpuno različite objekte. Pretpostavimo da imamo svojstvo pod nazivom *prazno* (engl. *empty*). U redu je da koristite svojstvo *prazno* kad god ono može da se primeni, pa možete da kažete da je mački prazan stomak i da je mačkina činija prazna.

*nastavak na sledećoj strani*

Uočite da tastatura računara i točkovi bicikla nisu smo svojstva; to su takođe i punopravni objekti, koji mogu imati sopstvena svojstva. Prema tome, objekti mogu imati podobjekte.

### Metode

Ono što objekti mogu da rade naziva se *metoda*. Mačke predu, računari se kvare, točkovi se kotrljaju. Objekti JavaScripta takođe imaju metode: dugmad `click()`, prozori `open()`, a tekst može da bude izabran `selected()`. Za-grade označavaju da je u pitanju metoda, a ne svojstvo.

#### ✓ Savet

- Može biti od pomoći ako objekte i svojstva posmatrate kao o imenice, a metode kao glagole. Prvo su stvari, a drugo aktivnosti koje te stvari mogu da izvršavaju, ili su na njima izvršene.

### Povezivanje delića

Objekte, svojstva i metode možete da povežete, kako biste dobili bolji opis nekog objekta, ili da biste opisali neki postupak. U JavaScriptu, oni su razdvojeni tačkama (kao u Internet adresama). Ovo se naziva *sintaksa tačaka*. Ovde su navedeni neki primeri objekata i njihovih tako napisanih svojstava:

```
bicycle.wheels  
cat.paws.front.left  
computer.disk.floppy  
document.images.name  
window.status
```

A ovde su neki primeri objekta i metoda napisanih u sintaksi tačaka:

```
cat.purr()  
document.write()  
forms.elements.radio.click()
```



Tabela 1.1

Upravljačka metoda događaja	
DOGAĐAJ	ČIME RUKUJE
onAbort	Korisnik je prekinuo učitavanje stranice
onBlur	Korisnik je napustio objekat
onChange	Korisnik je izmenio objekat
onClick	Korisnik je mišem izabrao neki objekat
onError	Skript je naišao na grešku
onFocus	Korisnik je aktivirao objekat
onLoad	Objekat je završio učitavanje
onMouseover	Kursor se pomera iznad nekog objekta
onMouseout	Kursor se sklonio sa objekta
onSelect	Korisnik je izabrao sadržaj nekog objekta
onSubmit	Korisnik je podneo obrazac
onUnload	Korisnik je napustio prozor

## Rukovanje događajima

*Događaji* su aktivnosti koje izvršavaju korisnici u toku posete stranici. Podnošenje nekog obrasca i pomeranje miša iznad neke slike predstavljaju dva primera događaja.

JavaScript se bavi događajima koristeći komande pod nazivom *upravljačka metoda događaja* (engl. *event handlers*). Neka akcija korisnika na stranici pokreće upravljačku metodu događaja u vašem skriptu. Dvanaest najčešće korišćenih upravljačkih metoda događaja JavaScripta su navedeni u **tabeli 1.1**. O drugim, naprednijim, upravljačkim metodama događaja govoriće se u poglavlju 11.

Na primer, pretpostavimo da naša mačka rukuje događajem `onPetting` (milovanje) izvršavajući akcije `purr` (prede) i `stretch` (tegli se).

U JavaScriptu, ako korisnik pritisne neko dugme, upravljačka metoda događaja `onClick` zabeležiće tu akciju i izvršiti bilo koju dužnost koja mu je dodeljena.

Kada pišete neki skript, ne morate da predvidite sve moguće akcije koje korisnik može da izvrši, samo one gde želite da se nešto posebno dogodi. Na primer, vaša stranica će se sasvim dobro učitati bez upravljačka metoda događaja `onLoad`. Ali komandu `onLoad` ćete koristiti ako želite da aktivirate skript čim se stranica učita.

## Vrednosti i promenljive

U JavaScriptu, delić informacije predstavlja neku *vrednost*. Postoje različite vrste vrednosti; najpoznatije su brojevi. Vrednost *niza znakova* (*stringa*) jeste reč ili reči uokvirene znacima navoda. Ostale vrste vrednosti u JavaScriptu navedene su u **tabeli 1.2**.

*Promenljive* sadrže vrednosti. Na primer, promenljivoj `myName` dodeljen je niz znakova „Dori.” Drugi način da se ovo napiše jeste `myName="Dori"`. Znak jednakosti može da se pročita kao „dodeljena je vrednost” Drugim rečima, promenljiva `myName` sada sadrži vrednost „Dori.”

### ✓ Saveti

- JavaScript razlikuje velika i mala slova. To znači da `myname` nije isto što i `myName`, niti je isto što i `MyName`.
- Imena promenljivih ne mogu da sadrže razmake ili druge znake interpunkcije, ili da započnu brojem. One takođe ne mogu da koriste rezervisane reči JavaScripta. Listu rezervisanih reči JavaScripta možete naći u dodatku B.

## Operatori

Operatori su simboli koji se koriste u radu sa promenljivim. Već su vam poznati operatori iz obične aritmetike; plus i minus su operatori. Potpuna lista operatora data je u **tabeli 1.3**.

### ✓ Saveti

- Mada oba izraza `x++` i `++x` dodaju jedinicu promenljivoj `x`, oni nisu identični; u prvom izrazu je `x` uvećano pošto je vrednost dodeljena, a u drugom pre dodele vrednosti. Na primer, ako je `x` jednako 5 i `y=x++`, znači da je promenljivoj `y` dodeljena vrednost 5, a promenljivoj `x` vrednost 6, dok izraz `y=++x` dodeljuje vrednost 6 i za `x` i za `y`. Operator `-` (znak minus) radi slično.
- Ukoliko kombinujete vrednosti brojeva i nizova znakova pri sabiranju dve vrednosti, rezultat je niz znakova. Na primer, `cat + 5` kao rezultat daje `cat5`.

Tabela 1.2

Tipovi vrednosti		
TIP	OPIS	PRIMER
Number	Bilo koja numerička vrednost	3.141592654
String	Znaci između znakova navoda	„Hello, world!”
Boolean	True ili False	true
Null	Prazan i bez značenja	
Object	Bilo koja vrednost dodeljena određenom objektu	
Function	Vrednost koju vraća neka funkcija	

Tabela 1.3

Operatori	
OPERATOR	ŠTA ON RADI
<code>x + y</code> (Numeric)	Sabira <code>x</code> i <code>y</code>
<code>x + y</code> (String)	Povezuje <code>x</code> i <code>y</code>
<code>x - y</code>	Oduzima <code>y</code> od <code>x</code>
<code>x * y</code>	Množi <code>x</code> i <code>y</code>
<code>x / y</code>	Deli <code>x</code> sa <code>y</code>
<code>x % y</code>	Moduo od <code>x</code> i <code>y</code> (to jest ostatak pri deljenju <code>x</code> sa <code>y</code> )
<code>x++, ++x</code>	Dodaje jedinicu <code>x</code> (isto što i <code>x = x + 1</code> )
<code>x--, --x</code>	Oduzima jedinicu od <code>x</code> (isto što i <code>x = x - 1</code> )
<code>-x</code>	Menja znak <code>x</code>

Tabela 1.4

Dodeljivanje	
DODELA	ŠTA RADI
x = y	Dodeljuje x vrednost y
x += y	Isto što i x = x + y
x -= y	Isto što i x = x - y
x *= y	Isto što i x = x * y
x /= y	Isto što i x = x / y
x %= y	Isto što i x = x % y

Tabela 1.5

Upoređivanja	
UPOREĐIVANJA	ŠTA RADI
x == y	Vraća true ako su x i y jednaki
x != y	Vraća true ako x i y nisu jednaki
x > y	Vraća true ako je x veće od y
x >= y	Vraća true ako je x veće ili jednako y
x < y	Vraća true ako je x manje od y
x <= y	Vraća true ako je x manje ili jednako y
x && y	Vraća true ako su i x i y true
x    y	Vraća true ako je ili x, ili y, jednako true
!x	Vraća true ako je x jednako false

## Dodeljivanje i poređenje

Kada promenljivoj odredite vrednost, vi tu vrednost dodeljujete promenljivoj i koristite operator dodele da biste to uradili. Na primer, operator jednakosti koristite da biste izvršili dodeljivanje, kao što je `hi sName="Tom"`. Postoji čitav skup operatora dodele kao što je navedeno u **tabeli 1.4**.

Za razliku od znaka jednakosti, postoje operatori dodele koji služe kao prečice za promenu vrednosti promenljivih. Na primer, kraći način da se navede `x=x+5` jeste `x+=5`. U većem delu ove knjige koristili smo dužu verziju, radi lakšeg razumevanja.

### Poređenja

Često možete da poredite vrednost jedne promenljive sa drugom, ili vrednost neke promenljive sa vrednošću literala (to jest, sa vrednošću koja je uneta u neki izraz). Na primer, možda želite da poredite vrednost dana u nedelji sa „Tuesday,” a to možete da uradite proverom da li je `today'sDate=="Tuesday"`. Potpuna lista poređenja data je u **tabeli 1.5**.

#### ✓ Savet

- Ukoliko upoređujete nizove znakova, imajte na umu da je „a” veće od „A” a da je „abracadabra” manje od „be”.

## Koje alate koristiti?

Kako je JavaScript samo običan tekst, možete koristiti bilo koji program za uređivanje teksta. Možete čak da koristite i program za obradu teksta kao što je Microsoft Word, mada morate uvek da se uverite da je Word snimio datoteku kao Text Only, umesto u svom izvornom formatu za datoteke.

Bolje je da koristite programe koji kao svoj standardni format koriste običan tekst. Mnogi ljudi snalaze se u Windowsu koristeći program NotePad (**slika 1.5**). Ukoliko želite program koji je projektovan za pisanje HTML-om, dobar izbor predstavlja Macromedia HomeSite. Na Mac računarima možete koristiti TextEdit (na Mac OS X) ili SimpleText (u prethodnim verzijama Mac OS), mada je u profesionalnoj upotrebi omiljen BBEdit, Bare Bones Software (**slika 1.6**). Na Unix mašinama, jedan od najboljih raspoloživih programa za uređivanje teksta jeste Emacs. Bez obzira na to koji program koristite, ne zaboravite da snimate svoje datoteke kao datoteke sa običnim tekstom i ekstenzijom .html ili .htm kako bi sve išlo glatko kada te datoteke učitavate na neki veb server.

Možete da koristite i neke vizuelne, ili WYSIWYG (*What You See Is What You Get*) HTML programe za uređivanje teksta, kao što su Adobe GoLive, Microsoft FrontPage (za PC; ne postoji verzija za Mac) ili Macromedia Dreamweaver (koji sada poseduje mnoge funkcionalnosti HomeSite built-in). Samo se prebacite na njihove HTML Source režime, i nastavite da pišete skripte.



Slika 1.5 NotePad u Windows XP.



Slika 1.6 BBEdit u Mac OS X.