

Volume 13, Number 1, 2006

**ter**  
Technology Electronic Reviews

---

Formerly *Telecommunications Electronic Reviews*

Technology Electronic Reviews (TER) is a publication of the Library and Information Technology Association.

Technology Electronic Reviews (ISSN: 1533-9165) is a periodical copyright © 2006 by the American Library Association. Documents in this issue, subject to copyright by the American Library Association or by the authors of the documents, may be reproduced for noncommercial, educational, or scientific purposes granted by Sections 107 and 108 of the Copyright Revision Act of 1976, provided that the copyright statement and source for that material are clearly acknowledged and that the material is reproduced without alteration. None of these documents may be reproduced or adapted for commercial distribution without the prior written permission of the designated copyright holder for the specific documents.

---

## Contents:

- News from the *TER* Editorial Board. by Sharon Rankin.
- REVIEW OF: Helen R. Adams, Robert F. Bocher, Carol A. Gordon & Elizabeth Barry-Kessler. (2005). *Privacy in the 21st Century: Issues for Public, School, and Academic Libraries*. Westport, Conn.: Libraries Unlimited. (ISBN: 1-5915-8209-1) by Rob Withers.
- REVIEW OF: Jon Duckett. (2005). *Accessible XHTML and CSS Web Sites*. Indianapolis, IN: Wiley Publishing Inc. (ISBN: 0-7645-8306-9) by Craig S. Booher.
- REVIEW OF: Nigel MacFarlane. (2005). *Firefox Hacks*. O'Reilly and Associates: Sebastopol, CA. (ISBN: 0-596-00928-3) by Shirley Duglin Kennedy.
- REVIEW OF: Andrew Watt. (2005). *Beginning Regular Expressions*. Indianapolis, IN: Wiley Publishing Inc. (ISBN: 0-7645-7489-2) by Mark Cyzyk.
- REVIEW OF: Richard York. (2005). *Beginning CSS: Cascading Style Sheets for Web Design*. Indianapolis, IN: Wiley Publishing Inc. (ISBN:0-7645-7642-9) by David B. Malone.
- About TER

---

## News from the *TER* Editorial Board

by Sharon Rankin

After a four month hiatus, *TER* is back with five reviews for your reading pleasure, covering a number of topics that we hope you will find relevant relevant to your work and professional interests: library privacy issues, web design and accessibility, the Firefox browser and the use of regular expressions in programming code.

*TER*'s Publisher Relations Editor, Frank Cervone has been hard at work to prepare a new inventory to be reviewed. A call for reviewers will be made during the month of February.

If you cannot find a suitable title on the *TER* inventory, choose one of your library's titles and submit this suggestion. If the title is within scope, we would be very pleased to accept your review. For more information about becoming a *TER* reviewer Florence Tang, Reviewer Relations Editor at tang\_fy@mercer.edu. (mailto:tang\_fy@mercer.edu)

Laura Wrubel, *TER* Board member has prepared this issue, with lightening speed. Thank you to Laura for taking on web editing responsibilities.

If you have comments about this issue or have suggestions to make, do not hesitate to send to email to sharon.rankin@mcgill.ca (mailto:sharon.rankin@mcgill.ca).

---

**REVIEW OF: Helen R. Adams, Robert F. Bocher, Carol A. Gordon & Elizabeth Barry-Kessler. (2005). *Privacy in the 21st Century: Issues for Public, School, and Academic Libraries*. Westport, Conn.: Libraries Unlimited. (ISBN: 1-5915-8209-1)**

by Rob Withers

*Privacy in the 21st Century* is intended to explore current privacy issues in public, school, and academic libraries, with an emphasis on the impact that technology has on a library's ability to protect patron privacy. The core of the book consists of several chapters that address varying aspects of privacy: privacy in the United States, privacy and technology, and privacy and libraries. The book then continues with one chapter on each of the issues specific to public, school, and academic libraries. A concluding chapter, The Future of Privacy in Libraries, rounds out the book and discusses possible future developments and the ways in which librarianship should respond to them.

The authors collectively have a diverse background, including one author each with experience in school and public librarianship, one who is an education professor, one who is an attorney formerly employed by the Center for Democracy and Technology and currently employed by the Internet provider Earthlink. The book's forward is written by the Director of the American Library Association's Office for Intellectual Freedom Office, Judith Krug.

*Privacy in the 21st Century* is organized with the intent that readers can dip into whichever parts of the book are most relevant to their library setting, with the caveat in the book's introduction that some material may sometimes be repeated in multiple places.

Thankfully, the authors succeed in this venture. There are a few cross-references from one chapter to another, which helps to minimize redundancy and direct readers to more information on topics such as FERPA (Family Educational Rights and Privacy Act). While some topics, such as RFID (Radio Frequency Identification) technology or the USA-PATRIOT

Act, pop up in multiple places, they occur in reasonable enough contexts and quantities to avoid charges of recycling. In addition, the glossary at the end of the book will help readers to familiarize themselves with key jargon or terms, although from time to time, readers will have to rely on the text to pick up on a key term-- for example, PII -- personally identifying information.

The first chapters are a clear and readable primer of key legal and technological developments associated with privacy rights, particularly within the library realm, and should be sufficient to bring general readers up to speed.

Perhaps because of the amount of information being presented in a confined amount of space, there are a few instances when the book could have provided more. For example, when recommending Spycatcher software for detecting cookies and other potential threats to privacy, the book does not mention its reason for including only this software, nor does it provide information about other software options, or provide URL reference source for more information(p. 156). Omissions such as these are exceptions to a generally outstanding job of writing.

Addressing privacy issues that confront libraries can be challenging due to the complexity of both libraries and the law. As the authors note in their introduction, privacy laws can be a crazy quilt that varies on a state-by-state basis. Moreover, because interpretation of laws is sometimes unclear, privacy policies within individual organizations or institutions can come into play. While a small volume cannot hope to address every potential facet of privacy issues within such a framework, the authors consistently provide examples drawn from several locales. In some instances where laws or interpretations of them vary, multiple contrasting examples are presented. *Privacy in the 21st Century* avoids oversimplifying the sometimes difficult conundrums posed by current laws and technology and acknowledges ambiguity. Examples might include a discussion about what library records fall under the provisions of open records laws, which vary from one locality to another and be open to multiple interpretations.

*Privacy in the 21st Century* is generally careful to discuss pros and cons of solutions, or to explore more than one possible strategy for safeguarding patron's privacy. This approach is helpful in situations such as attempts to discourage access to pornography, where there are many solutions, each with its own set of challenges (p. 80).

In addition, while drawing heavily on many documents drafted by the American Library Association or its entities, it steers clear of simply parroting these documents or toeing the party line. In one instance, when discussing the American Library Association's Acceptable Use of the Internet Policy, it notes that the policy states that libraries should explicitly prohibit use of library material to access obscene material, but later states that when deciding on what is obscene, libraries and librarians are not in a position to make those decisions for end users (p. 86-87).

The inclusion of supplementary materials further strengthens *Privacy in the 21st Century*. One chapter is devoted to a listing of additional resources and readings although curiously, this chapter is placed just before the conclusion, rather than as an appendix, where it might be expected to appear. There are a series of appendices that contain core documents related to privacy in libraries, sample documents for a privacy audit, and sample policies, which can be helpful for those looking to respond to the issues raised by this book in individual libraries. The generous endnotes in each chapter provide even more recourse for pursuing issues and solutions explored in this text.

*Privacy in the 21st Century* is built around the assumption widely shared within librarianship that a minimum of intrusion into information use by others is ideal. This approach will work with the intended audience of those in the library community. Given the politicization of some aspects of privacy rights, one wishes that the book also included a chapter on why privacy rights are essential within the context of a 21st century democracy, and why some developments that large segments of the American public, for example the USA-PATRIOT Act, or CIPA (the Children's Internet Protection Act) threaten these rights. This might help librarians to frame this discussion with those outside the profession of librarianship, particularly to outside governing boards where some do not share these same predispositions.

Much of this book focuses on current and emerging information technologies and their potential to impact the privacy of library patrons. The authors are careful to explain how individual technologies work, how they have the potential to adversely impact technology, and what, within reason; libraries can do to minimize such an impact. As a result, the book should be understandable to all and provide a springboard for discussion between library administrators, information technology specialists, lawyers, and others.

*Rob Withers is Assistant to the Dean & University Librarian at Miami University Libraries in Oxford, Ohio.*

Copyright © 2006 by Rob Withers. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at [rwithers@lib.muohio.edu](mailto:rwithers@lib.muohio.edu) (<mailto:rwithers@lib.muohio.edu>).

---

**REVIEW OF: Jon Duckett. (2005). *Accessible XHTML and CSS Web Sites*. Indianapolis, IN: Wiley Publishing Inc. (ISBN: 0-7645-8306-9)**

by Craig S. Booher

Duckett addresses three main topics in this book: XHTML, CSS and accessibility. Obviously, each topic is sufficiently complex to warrant its own separate treatment. So how does the author cover all three in a standard 450-page work?

Quite simply, by limiting the scope of his discussion. Specifically, he assumes the reader is experienced at creating websites with HTML. The target reader will be familiar with versions of HTML used to create websites in the late 1990s. This book is intended to help the experienced Web site creator upgrade their skills to the tools of the new millennium, "to write Web pages that are attractive, accessible, and that conform to the new Web standards of XHTML and CSS" (p. xviii).

Published as part of the "Problem Design Solution" series, the book has an interesting teaching style. Chapter 1 introduces an imaginary commercial web site, First Promotions. Several primary pages from the site are depicted along with the HTML 3.2 markup used to create those pages. As the remainder of the book moves through the key topics, the author uses the First Promotions website to illustrate how the new technology can be used to create a site that is virtually identical in appearance, but which has a much more robust foundation and is considerably more user friendly.

Chapter 2 gently introduces the HTML-savvy reader to XHTML by way of XML. The fundamentals of XML are presented succinctly and relatively painlessly. The three versions of XHTML 1.0 (Strict XHTML 1.0, Transitional XHTML 1.0, and Frameset XHTML 1.0) are briefly described. Differences between HTML and XHTML are illustrated with sections describing the HTML 4.0 elements and attributes removed or changed to create Strict XHTML 1.0. The concept of validation is described and the reader is given a micro-snapshot of the process using Dreamweaver and the W3C validator. The chapter concludes with an extensive analysis of the changes that need to be made to the First Promotions website in order to make it XHTML-compliant.

The pattern for each of the subsequent chapters follows the model established in Chapter 2. First, the chapter topic is discussed. Then, the points made in the chapter are illustrated by applying them in the revision of the First Promotions site.

The next three chapters focus on the use of style sheets to improve the appearance of the XHTML-based pages created in Chapter 2. In addition to explaining the components of XHTML (extensible hypertext markup language), Chapter 2 also emphasizes the importance of removing presentation control from the markup language. Chapters 3, 4 and 5 show the reader how to re-establish that control using CSS (cascading style sheets).

Chapter 3 provides a good introduction to CSS, covering all the basics (including selectors, font properties, box properties and background/color properties) in a concise, logical flow. The section explaining the box model is particularly helpful. Chapter 4 covers pseudo-classes,

pseudo-elements, precedence of rules, modular style sheets and properties affecting presentation of links, tables and lists. Chapter 5 concludes this CSS exploration by describing the use of CSS, rather than tables, for layout and positioning of elements.

Chapters 6 and 7 cover accessibility, the third main focus of the book. People with disabilities face many different types of challenges when they seek to interact with websites. These users rely on assistive technologies to use the Web (and computers in general). Unfortunately, many websites (especially those that have not separated presentation from content markup) undermine the ability of these technologies to help the user interact with the site. Two primary guidelines for creating accessible websites have been published: the Web Content Accessibility Guidelines 1.0 and the Electronic and Information Technology Accessibility Standards (commonly referred to as the Section 508 guidelines).

Chapter 6 discusses many of the checkpoints and rules found in these two guidelines, illustrating for the reader the flaw in the standard HTML approach, the corresponding checkpoint that relates to this concern, and an explanation of how that checkpoint can be addressed. Chapter 7 expands the accessibility discussion to cover tables and forms, two common features of many websites (especially commercial sites) that are particularly challenging for readers with disabilities.

Several appendices provide useful information. One contains the final code for each of the First Promotions pages. This is also available from the publisher's website. Another lists XHTML elements while a third serves as a reference guide to a representative subset of CSS properties.

Although Emerson felt that "a foolish consistency was the hobgoblin of little minds," unnecessary inconsistencies can irritate if not confuse the reader exposed to them. This book has more than the average number of such head scratchers. Sometimes they were as simple as a spelling error (e.g., "juane" when the author meant "jaune" or "lable" instead of "label") and the less compulsive reader could easily move on.

The more dangerous kind typically manifested themselves in a discrepancy between the text and some sample code being discussed by the text. For example, when discussing style sheets in Chapter 3, the text refers to a "table cell [with] an id attribute whose value is siteDescription." The sample XHTML markup is correctly identified as "

---

...". However, when the next paragraph talks about the associated CSS rule, it assigns a value of "tagline" to the id selector in the sample, "td#tagline { ...". This sort of incongruity considerably lengthens, and potentially derails, the learning curve of the diligent reader. Some of these (including the example shown above) are included in the considerable listing of errata available from the [www.wrox.com](http://www.wrox.com) (<http://www.wrox.com>) Internet site; others are not. In short, the book could have benefited from a stronger editorial eye prior to publication.

The book's gutting and reconstruction of the First Promotions site provides the reader with tangible evidence that functionality and aesthetics do not need to be compromised in order to improve the accessibility of websites. Although the author had recommended actually editing the sample website's HTML files available from the publisher's Internet site, this reader found that simply studying the markup examples in the text was sufficient to gain an understanding of the principles being discussed. The book's well-designed format for illustrating the markup changes examined in the text provides a strong communication tool enabling the reader to rather easily follow the author's design plan as he explains the transformations he is making.

It is evident that Duckett cares about making websites accessible to users with disabilities. Even if you are already conversant with XHTML and CSS, the two chapters on accessibility are almost worth the price of the book. As a package, the book provides a practical introduction to all three topics for its intended audience, the experienced designer and builder of HTML-based websites.

*Craig S. Booher has over 20 years experience as an information professional designing, developing, and implementing information systems in academic libraries and corporate information centers.*

Copyright © 2006 by Craig S. Booher. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at [csbooher@athenet.net](mailto:csbooher@athenet.net) (<mailto:csbooher@athenet.net>) .

---

## **REVIEW OF: Nigel MacFarlane. (2005). *Firefox Hacks*. O'Reilly and Associates: Sebastopol, CA. (ISBN: 0-596-00928-3)**

by Shirley Duglin Kennedy

I dunno. Maybe it's just me, experiencing techno-burnout. But this book is way too much for moi.

I like Firefox a lot. I never bothered downloading and installing it when my primary home computer was a WinXP machine. Just lazy, I guess. But early this year, I bought an iBook. When I discovered that Safari, the Mac OS native browser, did not work with Blogger (as well as many other things), I decided to give Firefox a try. I'm glad I did. Now there is no going back. And I was, quite frankly, eager to get my hands on this book. But I'm having a heck of a lot of trouble getting into it.

This is not a book for the casual Web surfer or the Internet user who just wants something to work reliably and is not at all interested in poking around under the hood. Firefox is a very rich browser, in terms of features and possibilities for customization, and McFarlane -- a Mozilla/Firefox guru who died suddenly and prematurely this past June (<http://www.mozillazine.org/talkback.html?article=6842>) -- shows you how to squeeze every drop of extra utility out of the browser. If that is what you want to do.

I was able to claw my way through the first four chapters of the book Firefox Basics, Security, Installation, and Web Surfing Enhancements, although it got real geeky, real fast. It's good to know a bit about the installation and configuration processes for the browser (e.g., where it deposits its files, how to tweak its appearance, performance, etc.) to best suit your individual needs. The OS-agnostic among us will find it gratifying that the book covers numerous flavors of Windows, Mac OS, Linux and Unix.

Chapter 1 is an overview of the browser's features from an end-user perspective, including page display, navigation, performance, keyboard shortcuts, searching, etc.

Chapter 2 deals with security and is well worth your time, especially if you are responsible for computers in a public access environment. Firefox is a good choice for public computers; it's quite flexible in terms of how tightly it can be locked down, and is widely regarded as much more secure than Internet Explorer. Its open source nature means that if a security issue is discovered, it is promptly addressed by a worldwide development community. Covered here are password handling, controlling foreign code such as ActiveX embedded in web pages, automatic updates and locking down the browser to protect dummies (e.g., setting up security for non-technical users).

Of additional interest to those who may be nursemaiding public computers, Chapter 3 covers installation in depth, offering configuration suggestions for different types of users and environments. This is where configuration options, profiles, etc., are covered. You will learn where to find complimentary applications for the browser and how to get it ready for wide deployment across an entire facility as well as manage user profiles remotely.

Chapter 4, Web Surfing Enhancements will be of interest to serious Web users, as it discusses bookmark enhancements, tabbed browsing, control over image and ad displays, news and feeds, search plug-ins and extensions, etc. One hack shows you how to save lots and lots of web pages to your local disk without hassle: helpful if you do a lot of Web research.

After that, things became a little too intense for me. I'm much less involved with cutting edge technology in my job these days; I do very little Web development and don't use things like XML (except tangentially, as a consumer of RSS, etc.). Chapters 5 through 9 are definitely aimed at power users, web and application developers, and folks who want to create their own extensions and themes and otherwise make major changes to the way Firefox looks and

works. One of the cool things about an open source browser, of course, is that it is so malleable, and people with far more skill than I possess have developed add-ons for Firefox that have enhanced my own browsing experience.

Thus, some of the hacks further along in the book are worth a look (e.g., Hack #52: Stomp on Cookies, explains how to track, trap, configure, kill create, and otherwise diddle with HTTP cookies). And if you have the need, Hack #81 shows you how to hide a content filter in Firefox's core to make websites vanish like ninjas in the dark.

As with all O'Reilly and Associates books generally, this title is well-indexed. And you can see and try some sample hacks at the O'Reilly web page for this book:

<http://www.oreilly.com/catalog/firefoxhks/> (<http://www.oreilly.com/catalog/firefoxhks/>). By and large, all of the books in the Hacks series ( <http://hacks.oreilly.com/> (<http://hacks.oreilly.com/>)) are worth having around, even if the content is often beyond the level of the average Internet or computer user. Keep in mind that these are not books most folks will want to read cover-to-cover, but by browsing the tables of contents and indexes, you may be able to solve a specific problem that you or someone you work with is having.

*Shirl Kennedy is the base librarian at MacDill Air Force Base in Tampa, Florida. She writes the Internet Waves column for Information Today, and serves as deputy editor of ResourceShelf.com and editor of DocuTicker.com.*

Copyright © 2006 by Shirley Duglin Kennedy. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at [webdoyenne@gmail.com](mailto:webdoyenne@gmail.com) (<mailto:webdoyenne@gmail.com>).

---

**REVIEW OF: Andrew Watt. (2005). *Beginning Regular Expressions*. Indianapolis, IN: Wiley Publishing Inc. (ISBN: 0-7645-7489-2).**

by Mark Cyzyk

My programming career began with an attempt to compile a book-length bibliography of doctoral dissertations in philosophy. Such a large, textual work inevitably required substantial formatting and reformatting. In order to automate this process I learned all about the macro language supplied with WordPerfect 5.0 for DOS. It was simple, but powerful. As a programming novice at the time, the only routines I learned were simple search and replace functions. Running a very long macro on a very long piece of text on a very slow 8080 processor took a very long time, so I quickly learned that when searching and replacing

certain patterns of strings in a large textual document one must be careful. The machine, after all, is literal in a way that we humans typically are not. One of the great virtues of this book is that it points out the things that can go wrong when programmatically manipulating text via regular expressions and shows how those problems can be circumvented or solved.

But first, what is a regular expression? A "regular expression" is a pattern of characters in a text. It is "regular" insofar as it is identifiable and can therefore be used to match a pattern wherever it may occur throughout a text. Now, why the terminology here is denoted by the term "regular expression" instead of something more intuitive like "pattern matching" I don't know. But after all, we live in a technological world in which one of the key concepts of HTTP, the foundation of the World Wide Web, is spelled "referer". When confronted by the technical term "regular expression" I simply do an immediate internal translation to "pattern matching", and all is well.

This book consists of twenty-six chapters, the first ten illustrate the various aspects of regular expression technology, the remaining sixteen provide useful illustrations of how regular expressions are implemented in the top programming languages and applications. The core of the book is to be found in Chapters 3 through 9.

### Chapter 3: Simple Regular Expressions

This chapter introduces and illustrates how to write a regular expression for use in matching patterns in text where the pattern to be matched consists of single characters, multiple characters in a string, optional characters, as well as using special "metacharacters", characters that have a special meaning within a regular expression to specify such things as match a class of characters; match any character; match within a certain range of characters; match a pattern zero or more times; match a pattern one or more times; and match a pattern a specified minimum and maximum number of times within a string of text. This is an introductory chapter to these topics which sets the stage for the chapters that follow.

### Chapter 4: Metacharacters and Modifiers

A regular expression can be very simple, something for example like this: `cat`. This simple regular expression would match any sequence of a `c` followed by an `a` followed by a `t` in a string of text. So it would match that string as it appears in the words `cat`, `cats`, `catalog`, `catastrophe`, `concatenation`, etc. Metacharacters provide for more sophisticated pattern matching. Suppose for example you wanted to find all instances of `cat` or `cut` in a text. The period (`.`) metacharacter would allow you to do so: `c.t`. The period metacharacter matches any character. This chapter illustrates the use of the following metacharacters:

---

<code>.</code>	Matches any character
<code>\w</code>	Matches characters in the English alphabet, numeric characters, and the underscore
<code>\W</code>	Matches any character not matched by <code>\w</code>

<code>\d</code>	Matches a numeric digit
<code>\D</code>	Matches any character not matched by <code>\d</code>
<code>\s</code>	Matches a single space, a tab, or a newline character
<code>\S</code>	Matches any character not matched by <code>\s</code>
<code>\t</code>	Matches a single tab character
<code>\n</code>	Matches a single newline character escaped characters. Characters like the period and the backslash themselves must be "escaped" with a backslash: <code>\.</code> and <code>\\</code>

## Chapter 5: Character Classes

Character classes are convenient ways to match a range or entire classes of characters in a text. For instance:

---

<code>[a-z]</code>	Matches any single lowercase alphabetic character
<code>[a-z]+</code>	Matches any number of lowercase alphabetic characters
<code>[A-Z]</code>	Matches any single uppercase alphabetic character
<code>[0-9]</code>	Matches any numeric digit
<code>[aeiou]</code>	Matches any vowel

Parentheses can be used to group items:

`(33|44)` Matches either a 33 or a 44 in a text

Use of quantifiers specify how many characters to match:

`[aeiou]{2}` Matches any two consecutive vowels

Negated character classes can be specified with the metacharacter :

`[^a-z]` Matches any single character that is not a lowercase alphabetic

## Chapter 6: String, Line, and Word Boundaries

Metacharacter "anchors" are used to denote start and end of lines, and beginning and ends of words:

---

<code>^</code>	Matches the start of a line of text
<code>\$</code>	Matches the end of a line of text
<code>^\$</code>	Matches a blank line
<code>\b</code>	Matches either the beginning or the ending of a word
<code>\</code>	Matches the beginning of a word
<code>&lt;&amp;amp;amp;it;</code>	
<code>/&gt;</code>	

\> Matches the ending of a word

## Chapter 7: Parentheses in Regular Expressions

Just as one uses parentheses to group elements in a Boolean search query, so too are they used in regular expressions to group parts of the expression.

---

`(Doctor|Dr|Dr\.)` Matches "Doctor" "Dr" or "Dr."

Watt briefly discusses, in this chapter, the notion of "capturing" parentheses, something that strikes me as being extraordinarily powerful. Essentially, programming languages and applications that allow for capturing parenthesis will match what is inside the parentheses and assign what is found there to a numbered variable name. The example Watt gives is:

`(United) (States)`

as used against the text:

The United States

In this case, the first parenthetical expression will match the string "United" in the text and will assign the value found there to a variable named "\$1". The second parenthetical expression will match against the string "States" in the text and will assign the value found there to a variable named "\$2". These two variables are now available for use by the programmer.

## Chapter 8: Lookahead and Lookbehind

Another powerful feature of regular expressions is the ability to match a pattern based on a previously matched pattern or based on a pattern being followed by another pattern. This is called lookahead and lookbehind. There are several different flavors of these two features, but just as an example, here is what Watt uses to illustrate a "positive lookahead":

`Star(?:=Training)`

Basically this regular expression says "Find the string 'Star', but only if it is immediately followed by the string 'Training'".

Lookahead and lookbehind, with their variants, surely are powerful programming constructs.

## Chapter 9: Sensitivity and Specificity in Regular Expressions

The moral of this instructive chapter is: Sensitivity and specificity in a regular expression act in mutually opposing ways such that an increase in the sensitivity of an expression most often results in the decrease of specificity, and vice versa. So increasing either usually results in a tradeoff: The more sensitive your search, the more "false hits" are likely to result; the more specific your search, the more likely it will result in relevant items being excluded. Watt provides solid examples of this principle in action.

Chapters 11-26 discuss and illustrate the implementation of regular expressions in such applications, application programming platforms, and programming languages as: Microsoft Word; StarOffice/OpenOffice; findstr; PowerGREP; Microsoft Excel; SQL Server 2000;

MySQL; Microsoft Access; JScript and Javascript; VBScript; Visual Basic .NET; C#; W3C XML Schema; Java; and Perl.

A chapter on regular expressions and Python would have been nice, but I suppose the line had to be drawn somewhere and an already very large book (742 pages) brought to an end. The other thing I would have liked to see is the provision of an Appendix briefly summarizing the syntax of regular expressions, a visual summary of the first several chapters -- sort of a Quick Lookup area of the book that could be easily accessible to the working programmer.

Overall, the book is informative, comprehensive, well-written, and is therefore easily readable. The examples are well thought out and clear. I think this book is ideal for someone starting out with regular expression programming and it also would have great value to the programmer who, for instance, needs to switch programming languages, from, say, Java or Perl to PHP.

In the end it's all about solving programming problems. Suppose, for instance, you are given a file of text consisting of what appears to be IP addresses, each on its own line, but you need to match just the strings that actually are valid IPs and weed out the rest. How to do this?

It's simple - just match on the following regular expression!:

```
^((25[0-5]|2[0-4][0-9]|1[0-9][0-9]|[1-9][0-9]|[0-9])\.)\.{3}(24[0-5]|2[0-4][0-9]|1[0-9][0-9]|[1-9][0-9]|[0-9])$
```

*Mark Cyzyk, formerly Head of Information Technology in the Albert S. Cook Library at Towson University, is currently the Web Architect at Johns Hopkins University in Baltimore, Maryland.*

Copyright © 2006 by Mark Cyzyk. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at [mcyzyk@jhu.edu](mailto:mcyzyk@jhu.edu) (<mailto:mcyzyk@jhu.edu>).

---

**REVIEW OF: Richard York. (2005). *Beginning CSS: Cascading Style Sheets for Web Design*. Indianapolis, IN : Wiley Publishing Inc. (ISBN:0-7645-7642-9)**

by David B. Malone

Richard York's work is well thought out and benefits greatly from a useful layout that includes numerous screenshots and portions of actual code. This volume is very practical for beginners, but can also be helpful for more experienced designers. An interesting feature of

York's text is the intentional pedagogical approach. York provides definitions for new terms as they are introduced and provides a context for better understanding. He suggests mnemonics to aid the reader in grasping new concepts. Nearly every chapter is punctuated with a summary and a set of exercises for the reader to utilize to reinforce the content of the chapter. Answers to the exercises are provided in one of the very useful appendices, which also include a cascading style sheets (CSS) quick reference guide and other coding tools. Roughly, the first half of the book introduces CSS and its syntax while the remainder highlights the variety of properties that can be used to implement CSS.

Interestingly, hypertext markup language (HTML), a sublanguage of the larger standard generalized markup language (SGML) family, was created as a method to reflect the structural elements of a document-never intending to serve the layout or design needs of a webpage. This mixed-up nature shows the shift between a textual, or word-oriented, culture to a visual one. Since HTML doesn't have the capacity to manage the needs of a visual medium-the World Wide Web-some other mechanism is needed. CSS provides the ability of adding a few simple strings of code that can dramatically change the look and feel of a webpage. This is where CSS enters the alphabet-soup of the Internet.

Another element of York's encouraging beginners towards the future of web design is his focus on combining CSS with the emerging extensible hypertext markup language (XHTML) and the broader extensible markup language (XML). The Web, in its fullest sense and within particular websites, continues to grow at astounding rates. York shows how CSS can provide the means of managing the display of variety web content in a centralized manner. Though not included on a CD-ROM within the book itself, the thousands of lines of CSS code provided within the book are available for download from the publisher. The text would prove more appealing, however, if the code were included; especially in light of the extra steps needed to download the source code and given that the notice of the code's availability is buried within York's beneficial introduction.

With increasing access to the Internet and a wider range of Internet users, York addresses disability issues, particularly vision impairments, and relates their impact on CSS and web design as he encourages designers to seek greater accessibility. And related, cellular telephone usage is expanding worldwide and the growth in web-enabled telephony is increasing. Recognizing this, York is forward thinking in providing direction to developers in addressing the needs of handheld users. Additionally, in a world of numerous web browsers, York delves into the sticky world of cross-browser compatibility. Other compatibility issues revolve around addressing failures of properly rendered CSS in Microsoft's Internet Explorer-the browser with the largest installed base.

For the beginning web developer or the experienced coder, York's work provides easy access to a wealth of information regarding the benefits and implementation of cascading style sheets. He doesn't simply present a litany of code, but provides some theoretical issues

to consider, thus creating a volume that is a useful guide to an important aspect of web design.

*David B. Malone is Head, Archives and Special Collections at the Buswell Memorial Library, Wheaton College, Illinois.*

Copyright © 2006 by David B. Malone. This document may be reproduced inwhole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at David.B.Malone@wheaton.edu (mailto:David.B.Malone@wheaton.edu)

---

## **About TER**

Editor is Sharon Rankin, McGill University ( sharon.rankin@mcgill.ca (mailto:sharon.rankin@mcgill.ca)).

### **Editorial Board:**

- Frank Cervone, Northwestern University ( f-cervone@northwestern.edu (mailto:f-cervone@northwestern.edu))
  - Tierney Morse McGill, Colorado State University ( tmcgill@manta.colostate.edu (mailto:tmcgill@manta.colostate.edu))
  - Florence Tang, Mercer University, Atlanta ( tang\_fy@mercer.edu (mailto:tang\_fy@mercer.edu))
  - Laura Wrubel, University of Maryland ( lwrubel@umd.edu (mailto:lwrubel@umd.edu))
  - Michael Yukin, University of Nevada, Las Vegas ( michael.yunkin@cmail.nevada.edu (mailto:michael.yunkin@cmail.nevada.edu))
-