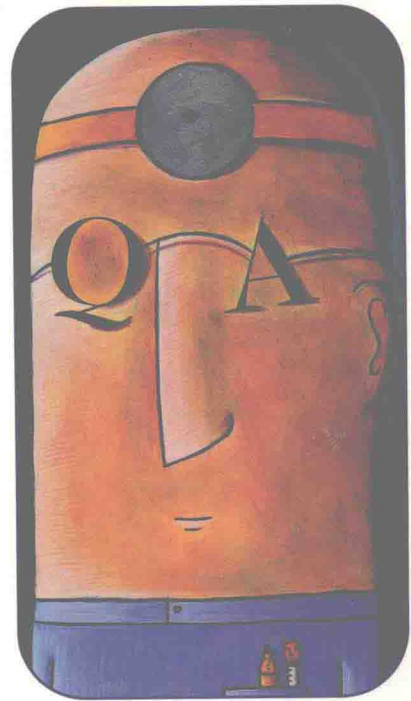


JAVA™



Developer's Resource

- ▶ Get started quickly with Java and object orientation
- ▶ Learn to use the AWT including Widgets, Windows, Menus, Layout Managers and more
- ▶ Master advanced features of Java including Threads, Exceptions, Streams, and Data Structures
- ▶ Put it all together in a new, extensible package for combining Java and VRML



A TUTORIAL AND ON-LINE SUPPLEMENT

Elliotte Rusty Harold

JAVA™

Developer's Resource

A TUTORIAL AND ON-LINE SUPPLEMENT



Prentice Hall PTR
Upper Saddle River, NJ 07458
<http://www.prenhall.com>

Elliotte Rusty Harold

Library of Congress Cataloging-in-Publication Data

Harold, Elliotte Rusty,

Java developer's resource : a tutorial and on-line supplement /

Elliotte Rusty Harold.

p. cm.

Includes index.

ISBN 0-13-570789-7 (alk. paper)

1. Java (computer program language) I. Title.

QA76.73.J38H37 1996

005.13'3--dc20

96-26213

CIP

Editorial/Production Supervision: Craig Little

Acquisitions Editor: Mary Franz

Editorial Assistant: Noreen Regina

Buyer: Alexis R. Heydt

Cover Design: Anthony Gemmellaro

Cover Design Direction: Jerry Votta

Art Director: Gail Cocker-Bogusz

Series Design: Meg VanArsdale



© 1997 Prentice Hall PTR

Prentice-Hall, Inc.

A Simon & Schuster Company

Upper Saddle River, NJ 07458

All rights reserved. No part of this book may be reproduced, in any form or by any means, without permission in writing from the publisher

The publisher offers discounts on this book when ordered in bulk quantities.
For more information, contact

Corporate Sales Department,

Prentice Hall PTR

One Lake Street

Upper Saddle River, NJ 07458

Phone: 800-382-3419; FAX: 201-236-7141

E-mail (Internet): corpsales@prehall.com

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN 0-13-570789-7

Prentice-Hall International (UK) Limited, *London*

Prentice-Hall of Australia Pty. Limited, *Sydney*

Prentice-Hall Canada Inc., *Toronto*

Prentice-Hall Hispanoamericana, S.A., *Mexico*

Prentice-Hall of India Private Limited, *New Delhi*

Prentice-Hall of Japan, Inc., *Tokyo*

Simon & Schuster Asia Pte. Ltd., *Singapore*

Editora Prentice-Hall do Brasil, Ltda., *Rio de Janeiro*

JAVA™

 Developer's Resource

Where's the CD?

Java's evolving so fast that what we could put on a CD today would be out of date tomorrow.

Instead, we've integrated the book with an On-Line Supplement: an up-to-the-minute Web site with a Java Newsletter direct from the author, answers to the problems in this book, Brewing Java tutorial, code from the book, the Java FAQ, Java challenges, and links to key Java sites.

Point your browser to

http://www.prenhall.com/developers_resource_series

To my wife, Beth

Trademarks

Adobe is a registered trademark of Adobe Systems Incorporated.

Apple, AppleScript, HyperCard, Macintosh, PowerBook, and the Apple logo are registered trademarks and Mac, MacOS, Chicago, Geneva, and System 7.0 are trademarks of Apple Computer, Inc.

Digital, DEC, ULTRIX, VAX, and VMS are registered trademarks of Digital Equipment Corporation.

Informix is a registered trademark of Informix Software, Inc.

IBM, AIX, OS/2, and PowerPC are registered trademarks of International Business Machines Corporation.

Hewlett-Packard, HP, and HP-UX are registered trademarks of Hewlett-Packard, Inc.

Helvetica and Times are registered trademarks of Linotype AG.

Metrowerks is a registered trademark and CodeWarrior is a trademark of Metrowerks.

Microsoft and MS-DOS are registered trademarks and Windows 95, Windows NT, and ActiveX are trademarks of Microsoft Corporation.

MIPS is a registered trademark of MIPS Computer Systems, Inc.

Arial and Times New Roman are registered trademarks of Monotype Corporation PLC.

Roaster is a registered trademark of Natural Intelligence, Inc.

Netscape Navigator is a registered trademark of Netscape Communications Corporation.

Nisus is a registered trademark of Nisus Software.

Motif, OSF, and OSF/1 are registered trademarks of the Open Software Foundation.

Oracle is a registered trademark of Oracle Corporation.

Quarterdeck is a registered trademark and WebStar is a trademark of Quarterdeck Corporation.

IRIX is a registered trademark of Silicon Graphics Inc.

Java, HotJava, Java Developer's Kit, SPARC, SunOS, and SunSoft are registered trademarks and Sun Microsystems, Solaris, and NFS are trademarks of Sun Microsystems, Inc.

Café and Café Lite are registered trademarks of Symantec Corporation.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Ltd.

All other product names mentioned herein are the trademarks of their respective owners.




CONTENTS


Preface 1

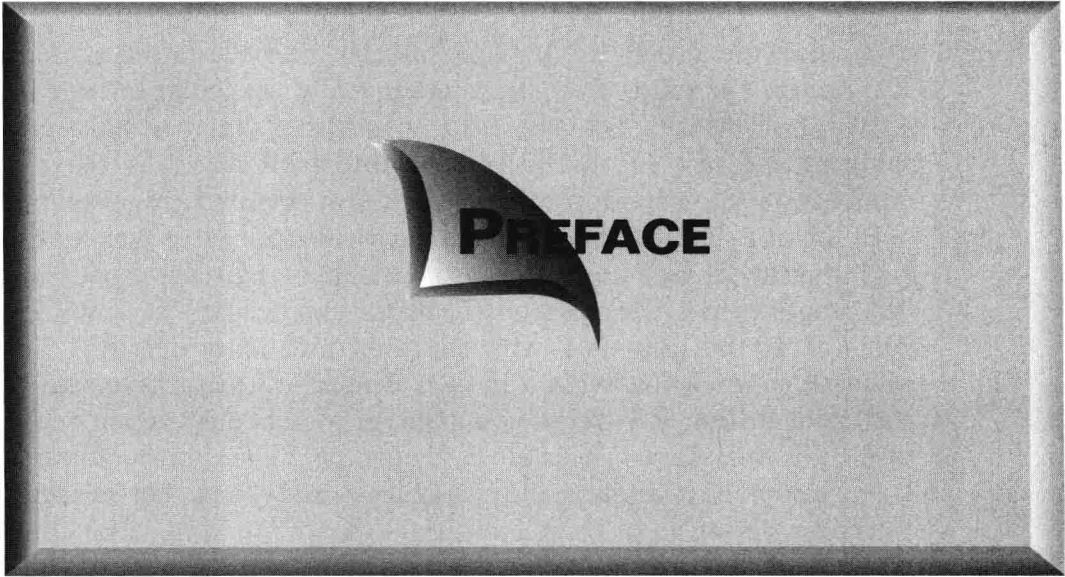


Part One: Getting Started with Java 6

CHAPTER 1	Why Java's Hot	9
CHAPTER 2	Hello World	35
CHAPTER 3	Extending Hello World	65
CHAPTER 4	Numbers and Arithmetic	89
CHAPTER 5	Booleans and Flow Control	113
CHAPTER 6	Intro to OOP	141
CHAPTER 7	Polymorphism and Inheritance	169
CHAPTER 8	Packages	189
CHAPTER 9	Arrays and Vectors	207

	Part Two: Applets and the AWT	226
CHAPTER 10	Writing Your First Applet	229
CHAPTER 11	Drawing Pictures	253
CHAPTER 12	Rectangles, Ovals, Polygons, and Images	275
CHAPTER 13	Events	295
CHAPTER 14	Widget Wizardry	325
CHAPTER 15	More Widgets	341
CHAPTER 16	LayoutManagers	367
CHAPTER 17	Windows, Frames, Dialogs, and Menus	403

	Part Three: Advanced Language Features	424
CHAPTER 18	Exceptions	427
CHAPTER 19	Threads and Animation	447
CHAPTER 20	Streams and Files	473
CHAPTER 21	Data Structures and java.util	501
CHAPTER 22	javadoc	523
CHAPTER 23	Java and VRML	539
CHAPTER 24	Java Resources	585
	Index	597



Hello world. Welcome to the Java Developer's Resource. I hope you like it.

Java has caused more excitement than any development on the Internet since Mosaic. Internet newsgroups and mailing lists dedicated to Java generate hundreds of messages a day. Thousand-seat Java classes fill up within hours of announcement. Everyone, it seems, is talking about Java. Unfortunately, very few people seem to know anything about it. This book is designed to change that.

Who You Are

This is a book for serious programmers. If Jolt Cola flows in your veins and you think John Von Neumann, Alan Turing, and Grace Hopper are some of the most important figures of this century, then this book is for you. I've never been fond of books that talked down to me, so I hope I've managed to avoid that flaw here.

Nonetheless, it's impossible for authors to know exactly where every one of their readers will be coming from. Some of you, I'm sure, write super-efficient hash table algorithms in C++ on napkins at breakfast while others may be spending their days writing button

handlers in HyperTalk. Some of you are spectacular Macintosh programmers while others think graphical user interfaces are a waste of CPU cycles. Therefore, I try not to assume any particular knowledge in this book. Most especially, I do not assume that you have programmed in C or C++, that you have written software for GUI environments, or that you are familiar with object-oriented programming.

On the other hand, I do hope you have some prior experience with programming. There's no question that the more experience you have with programming, the easier it will be for you to learn Java. While I don't assume that you know what the syntax of a for loop is in C, I do assume that you know what a loop is. Similarly, I assume you know what an algorithm is and how you might go about constructing one to solve a given problem. These are concepts that should be common to all programmers, regardless of the languages or systems they're accustomed to working with. There is a certain way of thinking, of planning an attack on a problem, that comes with being a programmer, just as there is a way of thinking that comes with being an artist or a journalist or a scientist; and it is this style of thinking that I assume.

If this isn't true, if you have no prior programming experience at all, I can still promise you that Java is an easier language to teach and learn than most, and you could certainly do worse than trying to learn the skill of programming with Java and this book. Intelligent people have certainly mastered programming beginning with far worse books and infinitely more complex languages. Nonetheless, programming is a skill that's difficult to teach without direct, personal interaction between human beings. I certainly would never have learned to program without teachers to help me over the initial rough spots. If you have no prior programming experience at all, I recommend taking a short course in Java programming to get up and running on the basics of writing and debugging programs, and, more importantly, to learn how programmers think.

Beyond programming skills, this book assumes that you are familiar with the Internet and the World Wide Web and have at least a rudimentary knowledge of HTML. You don't need to be an HTML expert, but you should be able to create a basic web page including links, style tags, and images. If your HTML needs some brushing up, I recommend Larry Aronson's *The HTML 3 Manual of Style* (Ziff-Davis, 1995).

This book also assumes familiarity with high-school-level math. Do you remember asking your teacher what algebra was good for? Computer programming (in any language, not just Java) is one of the things it's good for. It's impossible to teach programming without

some familiarity with basic math. However, the math used here is fairly elementary and should not be a large burden for most people.

Using This Book

This book is designed to be read pretty much from start to finish. While it is occasionally possible to read chapters out of order (for example, Chapter 18 on Exceptions can be read any time after Chapter 9), I think you'll get the most out of this book if you read it in the intended order.

Some readers, especially those with a great deal of experience in C++, may find a few of the sections overly obvious, particularly Chapters 6 and 7, which introduce object-oriented concepts for the first time. While I've tried to spice things up for everyone with a few well-placed anecdotes, you should feel free to skip ahead once you understand the syntax and purpose of a particular Java feature.

Chapter Enders

You will find five sections at the end of each chapter. First is a Q&A section with some more detailed questions that may have arisen in the minds of more experienced readers or those of you with particular backgrounds. Then there's a quiz. Everyone should consider the quiz questions. Most of which are a little on the devious side. I hope you'll find them enjoyable and not a little challenging. Third is a series of exercises. For the most part, these range from very simple to quite complex. You may pick those that seem most interesting to you, but it is important that you at least attempt some of them. You can only learn programming by writing programs, not just by reading a book. Then I give you a brief summary of what you should have gleaned from the chapter, a preview of what's coming up, and finally some suggestions for further reading.

Style

A few notes about the style. I've used a source code coloring editor to delineate different parts of Java programs. Java keywords are in bold. Comments are in italic. When entering programs, you do not need to use this formatting. The compiler cares only about the text. It does not concern itself with the formatting.

The appropriate grammar for a book about programming is always a little questionable, especially in a case-sensitive language like Java.

When the rules of English have conflicted with the rules of Java, I have regrettably let English be the loser. You will note that on occasion sentences do not always begin with capital letters. It has sometimes been necessary to begin a sentence with a Java word or phrase that simply does not mean the same thing when capitalized. Most of the time this should be obvious from the context and the formatting.

Conversely, Java tends to capitalize names of classes; and many words that you may not normally think of as proper nouns, like `Component` and `String`, are in fact proper nouns in the context of Java. Therefore, they have been capitalized when they are used to refer to their specific use in Java rather than their more generic meaning.

Similarly, it may appear that spaces are missing in a word like `LayoutManager`. Again, this is because that word needs to fit into Java's rules for naming classes before it fits into the Queen's English. Most of this should be obvious from the context.

Cross-Platform Issues

I am quite pleased to report that almost all the programs in this book have been tested on Solaris, Windows 95, and the Macintosh. At least for simple programs of the sort used for teaching, Java's claims of cross-platform compatibility seem to be accurate. You will note that screen shots alternate between all three platforms.

In fact the largest problem I encountered in trying to make this book platform-neutral was accounting for different terminologies on different platforms, a problem endemic to any cross-platform book. For the record, Mac users should read `folder` wherever they see the word `directory`, and Windows and UNIX mavens should reverse this.

Additional Resources

When I started work on this book many months ago, I posted some of my notes on a web site called *Café Au Lait*, which currently resides at <http://sunsite.unc.edu/javafaq/>. As preliminary and incomplete as those notes were, they attracted the attention of Mary Franz at Prentice Hall and led to the book you have in front of you. I hope I know a little more about Java now, and I hope I'm able to impart some of that knowledge to you.

The original *Café Au Lait* web site is still active and has continued to grow, and I encourage you to visit (<http://sunsite.unc.edu/javafaq/>). Among other things, it now includes the source code from this book

and answers to the various exercises and quizzes. It also includes a preliminary version of this book, a Java FAQ list, breaking news about Java, a list of Java user groups, and the usual collection of links to other sites.

As well as the web site, I also run a low-volume mailing list called Café Au Lait which includes various of my musings about Java and other matters. To subscribe, send email to `listproc@educom.unc.edu` with the following text in the body of the message:

```
subscribe cafeaulait Your_Full_Name_Here
```

To unsubscribe repeat these instructions but replace “subscribe” with “unsubscribe.” This is a low-volume, one-way mailing list that should not clog anyone’s mailbox. Back issues are available on the Café Au Lait web site.

I am interested in hearing from readers. You can correspond with me directly via email to `elharo@sunsite.unc.edu`. I am especially interested in hearing about mistakes in this book, and parts you may find less than clear. The more interesting queries may be answered in my newsletter, or in future editions of this or other books. However, please realize that I get several hundred email messages a day; and I cannot and will not respond to personal requests for help. If you have a question about Java, I recommend posting to the appropriate `comp.lang.java` newsgroup, where it will be seen by thousands of people able to help you.

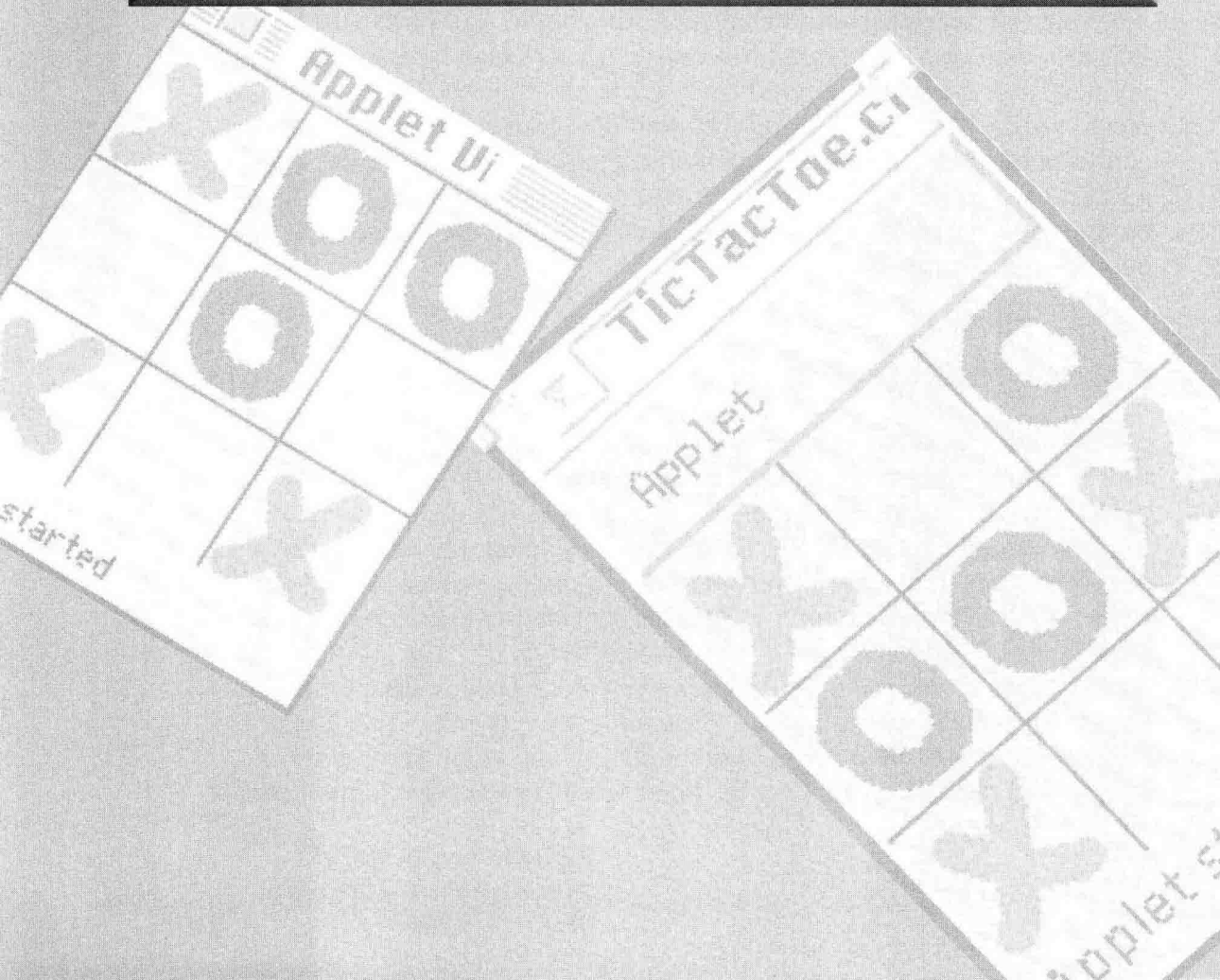
Acknowledgements

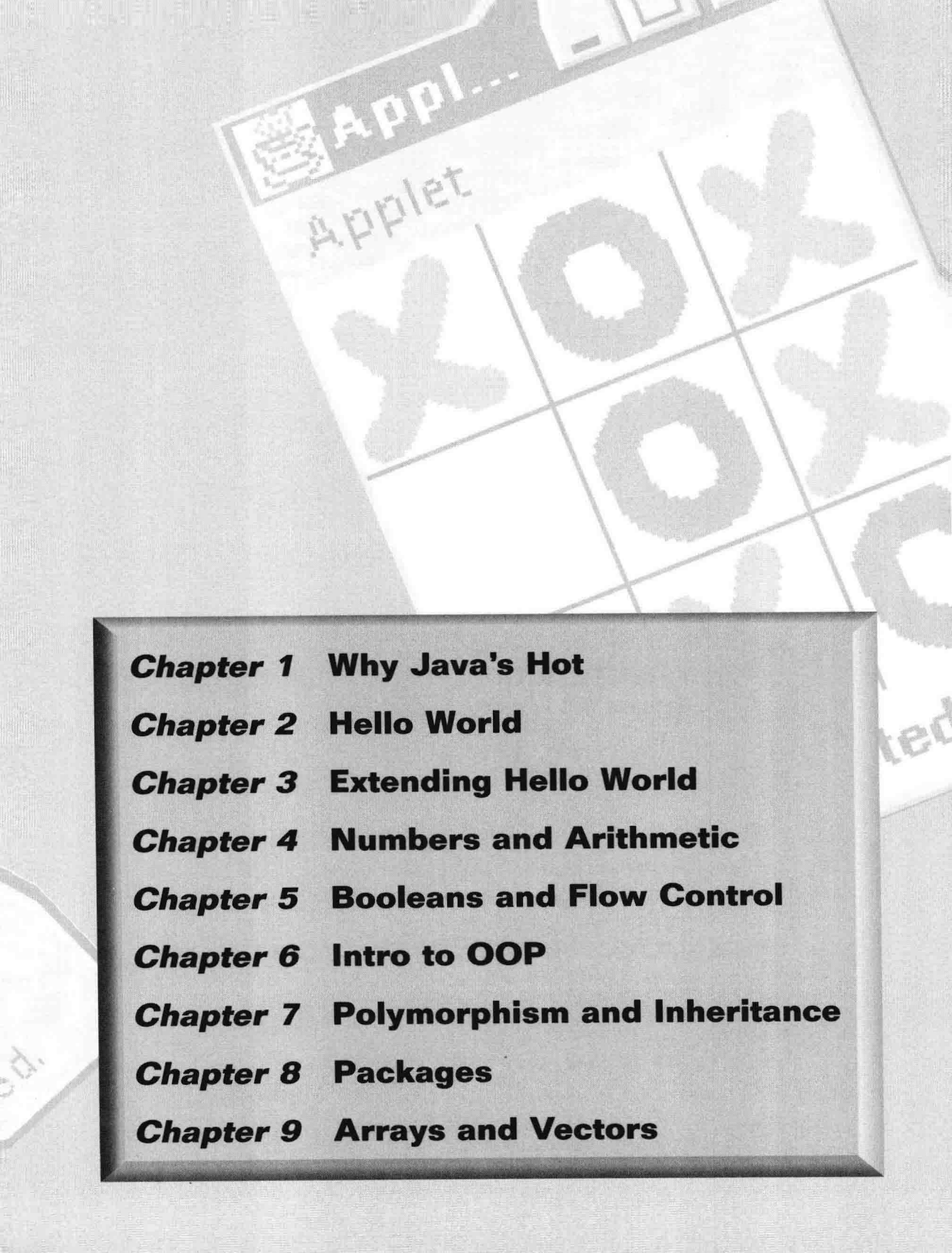
Many people were involved in the production of this book. Donald Ball of `sunsite.unc.edu` gave the original Web version of this book a home where it could be spotted by my publisher. My acquisitions editor, Mary Franz, got this book rolling; and my agent, David Rogelberg convinced me it was possible to make a living writing computer books. My development editor, Thomas Powell, provided many helpful comments that substantially improved the book. My production editor, Craig Little, put up with my insane attempt to write a book in ClarisWorks. All these people deserve much thanks and credit. Finally, I’d like to save my largest thanks for my wife, Beth, without whose support and assistance this book would never have happened.

Elliotte Rusty Harold
elharo@sunsite.unc.edu
June 24, 1996

PART ONE

Getting Started with Java



- 
- Chapter 1* Why Java's Hot**
- Chapter 2* Hello World**
- Chapter 3* Extending Hello World**
- Chapter 4* Numbers and Arithmetic**
- Chapter 5* Booleans and Flow Control**
- Chapter 6* Intro to OOP**
- Chapter 7* Polymorphism and Inheritance**
- Chapter 8* Packages**
- Chapter 9* Arrays and Vectors**

