CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

# *Java for the COBOL Programmer*

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

**Advances in Object Technology Series**
Dr. Richard S. Wiener, Series Editor
and Editor-in-Chief of
*Journal of Object-Oriented Programming*
SIGS Publications, Inc.
New York, New York

and

Department of Computer Science
University of Colorado
Colorado Springs, Colorado

1. Object Lessons: Lessons Learned in Object-Oriented Development Projects • Tom Love
2. Objectifying Real-Time Systems • John R. Ellis
3. Object Development Methods • edited by Andy Carmichael
4. Inside the Object Model: The Sensible Use of C++ • David M. Papurt
5. Using Motif with C++ • Daniel J. Bernstein
6. Using CRC Cards: An Informal Approach to Object-Oriented Development • Nancy M. Wilkinson
7. Rapid Software Development with Smalltalk • Mark Lorenz
8. Applying OMT: A Practical Step-By Step Guide to Using the Object Modeling Technique • Kurt W. Derr
9. The Smalltalk Developer's Guide to VisualWorks • Tim Howard
10. Objectifying Motif • Charles F. Bowman
11. Reliable Object-Oriented Software: Applying Analysis & Design • Ed Seidewitz & Mike Stark
12. Developing Visual Programming Applications Using Smalltalk • Michael Linderman
13. Object-Oriented COBOL • Edmund C. Arranga & Frank P. Coyle
14. Visual Object-Oriented Programming Using Delphi • Richard Wiener & Claude Wiatrowski
15. Object Modeling and Design Strategies: Tips and Techniques • Sanjiv Gossain
16. The VisualAge for Smalltalk Primer • Liwu Li
17. Java Programming by Example • Rajiv Sharma & Vivek Sharma
18. Rethinking Smart Objects: Building Artificial Intelligence with Objects • Daniel W. Rasmus
19. The Distributed Smalltalk Survival Guide • Terry Montlick
20. Java for the COBOL Programmer • E. Reed Doke, Ph.D. and Bill C. Hardgrave, Ph.D.

*Additional Volumes in Preparation*



© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

# *Java for the COBOL Programmer*

E. Reed Doke, Ph.D.
Bill C. Hardgrave, Ph.D.

CAMBRIDGE UNIVERSITY PRESS

SIGS BOOKS

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

PUBLISHED BY THE PRESS SYNDICATE OF THE UNIVERSITY OF CAMBRIDGE
The Pitt Building, Trumpington Street, Cambridge, United Kingdom

CAMBRIDGE UNIVERSITY PRESS
The Edinburgh Building, Cambridge CB2 2RU, UK
40 West 20th Street, New York, NY 10011-4211, USA
10 Stamford Road, Oakleigh, Melbourne 3166, Australia
Ruiz de Alarcón 13, 28014 Madrid, Spain
Dock House, The Waterfront, Cape Town 8001, South Africa

http://www.cambridge.org

Published in association with SIGS Books

© Cambridge University Press 1999

All rights reserved.

This book is in copyright. Subject to statutory exception
and to the provisions of the relevant collective licensing agreements,
no reproduction of any part may take place without
the written permission of Cambridge University Press.

Any product mentioned in this book may be a trademark of its company.

First published in 1999
Reprinted in 2000, 2001

Design and composition by Andrea Cammarata
Cover design by Tom Jezek

Printed in the United States of America

*A catalog record for this book is available from the British Library.*

*Library of Congress Cataloging-in-Publication Data is on record with the publisher.*

ISBN 0 521 65892 6 paperback

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

*To the millions of*
*COBOL programmers in the world.*

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

# Contents

**Foreword** xiii

**Preface** xv

**Acknowledgments** xvii

**Introduction** xix

**Chapter 1: Why You Should Learn Java**
- History and Overview of Java 3
- The Popularity of Java 5
- What Makes Java Different? 5
  - Java Is Simple 6
  - Java Is Object-Oriented 6
  - Java Is Portable 7
- Will Java Replace COBOL? 10
- The Goal of This Book 13
- Summary of Key Points 16
- Glossary 17
- Bibliography 19

**Chapter 2: An Introduction to Object-Oriented Programming** 21
- History of OO 22
- Objects 24
- Classes 25



© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information



Diagramming Classes and Instances 26
Inheritance 30
Class Relationships 34
Aggregation 34
Association 37
Object Communication 38
Polymorphism 41
Dynamic Binding 41
Summary of Key Points 43
Glossary 45
Bibliography 47

**Chapter 3: Java Structure** 49
A Class Program 50
Java Column Restrictions 55
Writing Comments in Java 55
Naming Rules and Conventions 57
Creating Instances 58
Invoking Methods 61
Working With Subclasses 66
Summary of Key Points 72
Glossary 74

**Chapter 4: Defining Data** 77
COBOL Picture Clause 78
Defining Java Variables 79
Writing Java Literals 81
The Scope of Variables 82
Defining Java Constants 83
String Variables 83

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information



Changing Variable Types 87
Variables for Community National Bank 89
Summary of Key Points 91
Glossary 93

**Chapter 5: Computation** 95
Exceptions 96
Custom Exception Classes 100
A Review of Primitive Data Types 105
Wrapper Classes 106
Arithmetic Operators 110
The Math Class 112
The NumberFormat Class 116
Summary of Key Points 120
Glossary 121

**Chapter 6: Decision Making** 123
The If Statement 125
Using the ELSE Clause 127
Nested IF Statements 129
Writing Compound Conditions 130
Java's Conditional Operator 130
Condition Names 131
Computing the Service Charge with if Statements 132
Case Structure: COBOL EVALUATE & Java switch 136
Computing the Service Charge Using switch 137
Summary of Key Points 141
Glossary 144

**Chapter 7: Loops** 145
The Loop Structure 146

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information



The COBOL PERFORM Statement 147
The Java while Statement 148
The Java do Statement 152
The Java for Statement 155
Nested Loops 158
Java break and continue Statements 160
Producing a Loan Amortization Schedule 161
Summary of Key Points 165
Glossary 166

**Chapter 8: Arrays** 167
Declaring One-Dimensional Arrays 168
Populating One-Dimensional Arrays 171
Creating String Arrays 173
Declaring Two-Dimensional Arrays 176
Populating Two-Dimensional Arrays 178
Passing Arrays as Arguments 182
Searching Arrays 183
Summary of Key Points 188
Glossary 189

**Chapter 9: Data Access** 191
Java's I-O Class Library (java.io) 192
Object Persistence 193
Sequential File I-O 194
Database Access 200
Object Serialization 207
Network Access 212
Summary of Key Points 214
Glossary 215

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information



**Chapter 10: Graphical User Interfaces** 217
Java's Abstract Windowing Toolkit 218
Event-driven Programming 218
Displaying & Closing a Window 220
Adding a Button to a Window 225
Adding Labels, TextFields, and Panels to a Window 231
Creating Pull-Down Menus 239
Writing Applets 243
Summary of Key Points 246
Glossary 248

**Chapter 11: Object-Oriented Development Issues** 249
Developing OO Systems 250
OO Methodologies 251
OO Analysis 252
OO Design 254
Three-Tier Design 256
Architecture Issues 266
Performance Issues 268
Summary of Key Points 270
Glossary 271
Bibliography 273

**Appendix: Programs on the CD-ROM** 275

**Glossary** 344

**Bibliography** 353

**Index** 355

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

# Foreword

COBOL programmers are, above all else, practical. They have seen languages come and go, each providing 15 megabytes of fame before vanishing in the night with little fanfare and less regret. COBOL programmers, meanwhile, quietly go about the business of developing and delivering the systems essential to business. Why then, given the hype and history of broken language promises, should COBOL programmers care about a language that more has been written about, than in?

Surprisingly, in many respects Java is much like COBOL. Each was defined with a very specific role to play, each eschews complexity for the sake of productivity, each strives for openness, and each is being defined by ISO to protect and promote corporate as well as personal investments in the language. Yet it takes more for a language to grow and prosper—it takes the right problem, or stated in slightly different terms—the right *opportunity*. The Internet is the opportunity for Java to prove its mettle.

Learning Java means many things. It does not mean abandoning COBOL, which will continue to thrive in the next century. The size of the information technology pie is exploding (and by pie we mean the business tasks, both existing and new, that require automation). We have only scratched the digital surface in terms of providing the systems that the company of tomorrow will demand. Our obsession with focusing inward will dramatically change to an outwardly directed view of the world. This fundamental shift will take us from systems integration to business integration. Building the infrastructure, constructing the numerous connections, and doing it right will require many sharp tools. COBOL and Java will serve complementary roles. COBOL excels in some areas, Java in others. The business/software engineers of today and tomorrow will exploit the best of both to make the age of information a reality.

First, learning Java means you will be exposed to a greater range of programming styles and new and exciting patterns of thought. A larger vocabulary of programming ideas will always stand you in good stead. Second, in that circular reinforcing way that knowledge works, knowing



© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information



Java will make you a better COBOL programmer, and knowing COBOL will make you a better Java programmer. Third, learning Java will enhance your ability to pick the right tool for the right job. Fourth, you will be in a position to increase the value of legacy systems and you will increase your own value to a company. And fifth, Java is fun.

I have the pleasure of knowing both Reed Doke and Bill Hardgrave. They come from the world of COBOL. They both teach COBOL and have written an excellent book called *An Introduction to Object COBOL* (John Wiley & Sons, 1998). In a nutshell, they know their COBOL. From this position they are able to build a bridge between the two languages. *Java for the COBOL Programmer* is clear, crisp, and concise. Brimming with insight and presented in an easy-to-follow, straightforward manner any COBOL programmer will appreciate, Doke and Hardgrave make learning Java easy and fun. Do yourself a favor and get a copy of this book, and then sit down and enjoy a good read. You will learn a lot. I know I did.

Edmund C. Arranga
Editor-in-Chief, *The COBOL Report*

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

# Preface

If you are like most of us, you have probably been thinking about updating your technical skills. You have been hearing a lot about Java, object-oriented (OO) development, and Internet applications. These topics have been getting a tremendous amount of press lately. In the fanfare, you may have heard someone suggest that COBOL programmers will be obsolete and can't possibly make the switch to OO programming. Can this possibly be true?

We wrote this book because we believe it is important that you learn Java and OO development. Although, we don't claim learning a new programming language is a trivial task, the fact that you already know COBOL gives you a head start on learning Java. Don't let what others say frighten you.

We work with COBOL as consultants for industry, in our classrooms, and as authors. However, we also work with Java and OO development. From our perspective, we believe COBOL and Java are highly complementary development tools in the evolving computing environment. COBOL does a super job of processing and maintaining a firm's data. Java plays an equally important role by capturing and reporting data by connecting clients to the server across a variety of networked computers, with little concern about the specific hardware and operating systems involved.

Today's business environment is diverse. A single platform and/or single language cannot meet the needs of most organizations. The business environment is becoming one of interconnectivity, requiring many different organizations with many different computing environments to communicate. Perhaps your organization is moving in this direction. Has your manager inquired about your interests in client-server computing or Java? The fact that you are reading this suggests you have been thinking about your future. We urge you to learn Java...not as a replacement for COBOL, but to make yourself even more valuable and make an even greater contribution to your firm's computing projects. Plus, we



© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information



find writing Java fun and we think you will too!

Many COBOL programmers are asking questions such as: What is Java? How does it differ from COBOL? How is it similar? What is the impact of Java on COBOL? What is object technology? We wrote this book to answer these questions for you.

Can you learn Java? Of course you can. Java has only 48 reserved words. At last count, COBOL had over 600! These numbers suggest the Java language set is much easier to learn and remember than many others, including COBOL, and our experience with Java corroborates this. Although it is not immediately obvious, there are many similarities between COBOL and Java. In fact, some of the statements are almost identical. We include a lot of Java code in the book and for most of it we have the corresponding COBOL code. Finally, you already know programming constructs – you need learn only the new syntax. But, you must also learn OO concepts and about half a dozen OO terms. Although not particularly difficult to learn, without these concepts, you will find it nearly impossible to learn Java. Therefore, before we get into programming statements, Chapter 2 explains OO for you in simple straightforward terms.

***Good luck, have fun,***
***and let us hear from you.***

Reed (ReedDoke@mail.smsu.edu)
Bill (whardgra@comp.uark.edu)

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

# Acknowledgments

We want to thank all of the people who assisted us in making this book a reality:

- Lothlórien Homet, the managing editor at SIGS Books, who kept all of us (mostly) on schedule.
- Edmund C. Arranga, President of Object-Z and editor of the *COBOL Report* for writing a very nice foreword and for his constructive comments about the approach and tone of the first few chapters.
- Dr. Neil Marple, technical area manager at SIAC in McLean, Virginia, for all of his encouragement and constructive suggestions, in addition to contributing a very sharp pencil to the technical editing process.
- Lisa McCumber, process specialist at Southwestern Bell in St. Louis for her astute and speedy technical review.
- Mike Hellebush, technical architect at Edward Jones in St. Louis, for providing a lot of real-world experience and insight, in addition to some technical editing chores.
- Dr. Lou Glorfeld at the University of Arkansas for some timely technical assistance with the database programs.



© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information

# Introduction

We organized the chapters in this book to be read in sequence. However, each chapter begins with a clear statement of what we assume you know before reading the chapter, so you can jump around a little bit, depending on your background and experience. You can use the book as a reference and jump in anywhere once you have the fundamentals.

Chapter 1 presents reasons why you should learn Java and describes the many similarities between COBOL and Java.

Chapter 2 explains what OO is, and what it is not. OO terms and concepts are described using several everyday examples.

Chapter 3 describes the overall structure and format of a Java program. Several small programs are developed to show you how to create object instances and call methods.

Chapter 4 shows you how to define Java data items and use them in a program. Java data definition is somewhat different than COBOL and these differences are clearly explained and demonstrated in the program examples.

Chapter 5 introduces Java computation and again, several small programs are written to illustrate the idea and concepts presented. You will see that some Java computation is nearly identical to COBOL.

Chapter 6 illustrates how to use the Java decision-making statements. We develop programs using the Java IF (sound familiar?) and the Java counterpart to the COBOL EVALUATE verb.

Chapter 7 describes how to write Java loops. As you will see, Java looping is different from COBOL.

Chapter 8 shows you how to define and work with Java arrays which are really the old COBOL tables with a more technical-sounding name. There are a lot of parallels between Java arrays and COBOL tables.

Chapter 9 explains how to access data in sequential files, relational databases and on networks. You will appreciate Java's approach to accessing relational databases using standard SQL statements.

Chapter 10 illustrates how to develop Graphical User Interfaces for



© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-65892-8 - Java for the COBOL Programmer
E. Reed Doke and Bill C. Hardgrave
Frontmatter
More information



user input and output. This is an interesting and important chapter, even though there are few COBOL similarities.

Chapter 11 discusses OO development in a broader context. Both software and hardware issues are explored. Three-tier software design is illustrated using a GUI front end and a relational database back end.

All code listings can be found on the Cambridge University Press Web site: *www.us.cambridge.org/titles/0521658926.html*

© Cambridge University Press www.cambridge.org