Java with Object-Oriented + Generic Programming

java

1.5.

Java with Object-Oriented + Generic Programming

ISBN 978-1-4276-3452-8

9 781427 634528 90000

by Paul S. Wang

sofpower.com

Java with Object-oriented and Generic Programming

Paul S. Wang

Department of Computer Science
Department of Mathematics and Computer Science
Kent State University
Kent, Ohio 44242-0001

Email: `pwang@cs.kent.edu`
Web: `http://java5.sofpower.com`
Publisher: `sofpower.com`
ISBN: 978-1-4276-3452-8

ISBN 978-1-4276-3452-8
90000
9 781427 634528

In Loving Memory of
My Mother

Chang Pei Wang
王張蓓

1921 – 2008

Preface

Since its beginning in the early 1990s, Java has grown and matured. Platform independence, object orientation, networking capabilities, graphical user interface (GUI) support and multithreading together make Java an ideal development tool for many applications. The last major revision came with the Java 5.0 (tiger) edition which introduced the *generics* for type-independent programming, `enum` and other improvements. This text helps you study these and other important aspects of Java so you can harness its full power.

Object-oriented design (OOD) and programming (OOP) concepts and techniques are integrated with the Java presentation so that you'll learn Java and object orientation at the same time. Topics are covered in detail and illustrated by realistic examples with object-oriented structures. The goal is to give a comprehensive and in-depth introduction to Java, OOP, and generic programming in a concise and practical manner.

The Book is Online

The book is a revised and updated version of *Java with Object-Oriented Programming*, 2nd edition, originally published by Thomson Learning. To reduce cost and make the material easily affordable, the text is available online in PDF format. You may purchase the entire book or individual chapters. The PDF files are easy to read and search on your computer where you can perform programming experiments immediately.

A companion Website provides the complete example *code package*, easy access to Java API documentation, hands-on activities, updates, and other useful resources for instructors and students. Examples in the book are backed by ready-to-run code in the code package and organized in directories corresponding to the chapters and major topic sections. Follow the user's guide in the code package to install it. Clearly marked examples in the book make correlating the descriptions and the actual code files easy.

Taking Advantage of PDF

With the PDF format, the book provides bookmarked table of contents, page thumbnails, and active links to figures, tables, and sections making it very convenient to use. Also, clicking on any index entry at the end of the book brings you directly to individual pages of interest.

In addition, you can scale the display up or down and search for any word or term in the entire book with Adobe Acroreader or some other comparable PDF display program. If and when needed, you can print hard copies of any desired pages easily.

Best of all, the PDF book is much less expensive than a traditional hardcopy book.

Up to Date

This text updates, expands, and reorganizes *Java with Object-Oriented Programming*, published in 2003 by Thomson Learning, to include new materials and emphasize OOP and generic programming. The companion Website offers information updates and hands-on experiments ready to use with the latest version of Java.

Reader, reader@school.edu

Object-oriented Programming

OOP is the dominant modern programming paradigm. Full and complete understanding of OOP is a must for any serious programmer. OOP concepts and techniques are explained thoroughly. The integration of theory with practice makes OOP principles concrete and provides reasons for Java constructs. Many complete examples demonstrate how different Java constructs and individual OOP techniques combine to achieve results in practice.

Java provides excellent support for OOP. But, simply using Java constructs does not automatically lead to well-organized object-oriented programs. On the contrary, without a good design created with an object-oriented view, the resulting program will most likely be a procedure-oriented program written in Java. Worse yet, it could be such a program bent out of shape to give rise to classes!

OO concepts and techniques are introduced early and integrated tightly with Java coverage. You'll learn OOP and practice it as you make progress through the text. Chapter 14 is devoted to (Object-oriented Design) OOD concepts, methods, and patterns. The OO design of major examples are shown with UML (Unified Modeling Language) class diagrams to reinforce object-oriented thinking.

Generic Programming

OOP enables us to write *polymorphic* programs that work for a hierarchy of related types. Generic programming aims to write completely type-independent (or generic) programs. Thus, a generic program will work on any given type.

There are two approaches for generic programming: the *catch-all-type* approach and the *type parameterization* approach. We present both approaches in depth and cover the new Java generics comprehensively.

GUI

Graphical user interfaces are great for making programs easy for end users. But GUI adds complexity and sophistication to a program. The Java Foundation Classes (JFC) provides an infrastructure and predefined components, or *widgets*, for GUIs. The Swing package is an important part of the JFC. Swing and its effective application are covered in-depth to get you started with GUI programming for *applets* and stand-alone applications alike.

Event-driven Programming

GUI requires a programming style that handles *events* at run-time. Instead of following a prescribed execution path, *event-driven programs* react to *external events* whose occurrence is unpredictable. Event-driven programming techniques are introduced and put to use in GUI and applet programming.

Comprehensive Coverage

Both basic and advanced topics are covered with emphasis on OOP and realistic applications.

- Basics—Java language constructs, classes, objects, using objects to solve problems, OOP tips, Java program structure, compilation, execution, error handling, and debugging.

- OOD and OOP—extending programs by inheritance, superclass and subclass, method overriding, plug-compatible objects, polymorphism, abstract superclasses, interfaces, uniform public interface planning, object cloning, iterators, design patterns, the model-view-controller (MVC) pattern in Java.
- Polymorphism, Generics and the Collections Framwork—writing and using programs that are type independent or applicable to a hierarchy of objects; the syntax and semantics of Java generics; the Java collections framework.
- GUI—using Swing widgets to construct graphical user interfaces for programs.
- Applets—writing and deploying Swing-based applets, the Java plug-in.
- Threads—concepts, techniques, and applications os multithreading. applications of multithreading.
- Advanced topics—networking with URL and sockets, writing server-side and client-side code, remote method invocation (RMI), multithreading, understanding concurrent programming and its challenges, mutual exclusion, scheduling, coordination of concurrent activities, animation as application of multithreading, classes as objects, introspection and reflection, dynamic loading of objects, interfacing to *native programs* (C, C++, fortran) through JNI on different platforms, the security manager, signed programs.

Examples illustrate concepts, constructs and usages, and show how different Java features combine to achieve useful purposes. Yet, comprehensive coverage, numerous full examples, good appendices, and a thorough index do not translate into a huge volume. In fact, the book is no thicker than a regular book.

Flexible Usage

The book is designed for a semester course at the upper college or beginning graduate level. Students should know C or C++ already. The book is self-contained, and it is entirely possible for a motivated student with less C/C++ knowledge to manage with extra effort. The book is suitable for an *Introduction to Java*, *Java with OOP*, or *Principles of OOP with Java* course. It can also be used as an auxiliary text for a *Theory of OOP*, *Networking*, *Graphical User Interface Design*, *Web Programming*, or *Concurrent/Parallel Programming* course.

The text can also be used for custom training courses for industry. A shorter course may omit chapters 10–12, as appropriate. An advanced course may assign most of chapter 1 and chapter 2 for reading and select more substantial programming projects from the exercises.

After a course, the text can be kept as a valuable reference.

Website

The text has a companion Website at

`http://java5.sofpower.com`

Information updates, on-Web examples, hands-on exercises, instructor notes (in PDF), articles, faq, links to documentation, and other resources can be found on the site.

Throughout the book, concepts and usages are thoroughly explained with examples. Instead of using contrived examples, however, every effort is made to give examples with practical value and to present them as complete programs ready to enter into the computer. The entire set of examples is available for download at the site.

Easy Reference

Information is organized and presented in a way that facilitates quick and easy reference. There are ample appendices including *Major Differences Between Java and ANSI C/C++* and *The Java Debugger:* `jdb`, a list of classes with section cross references, and a thorough and comprehensive index. This book will be a valuable aid for working with Java.

Acknowledgments

This online text evolved from the author's *Java with Object-Oriented Programming* book. It also benefited from two custom training courses conducted by the author at Goodyear Tire and Rubber, Akron, Ohio. Deep appreciation goes to my mother Chang Pei Wang, who passed away this year (2008), for her constant love and encouragement. Mom you will always live in my heart.

Finally my heart-felt thanks to my wife, Jennifer, and my children Laura, Deborah, and David for their support and encouragement.

Paul S. Wang
王 士 弘
Kent, Ohio