

Introduction

Some revolutions catch the world completely by surprise. The World Wide Web, the Linux operating system, and social networking all rose to prominence unexpectedly.

The remarkable success of the Java programming language, on the other hand, caught no one by surprise. Java has been the source of great expectations since its introduction a decade ago. When Sun Microsystems launched Java by incorporating it into web browsers, a torrent of publicity welcomed the arrival of the new language.

Sun cofounder Bill Joy didn't hedge his bets at all when describing the company's new language. "This represents the end result of nearly 15 years of trying to come up with a better programming language and environment for building simpler and more reliable software," he proclaimed.

In the ensuing years, Java lived up to a considerable amount of its hype. The language has become as strong a part of software development as the beverage of the same name. One kind of Java keeps programmers up nights. The other kind enables programmers to rest easier after they have developed their software.

Java was originally offered as a technology for enhancing websites with programs that run in web browsers. Today, it's more likely to be found on servers, driving dynamic web applications backed by relational databases on some of the Web's largest sites.

Each new release of Java strengthens its capabilities as a general-purpose programming language for environments other than a web browser. Today, Java is being put to use in desktop applications, Internet servers, personal digital assistants, embedded devices, and many other environments.

Now in its seventh major release—Java 6—the Java language has matured into a full-featured competitor to other general-purpose development languages, such as C++, Perl, Python, Ruby, and Visual Basic.

You might be familiar with Java programming tools, such as Eclipse, Borland JBuilder, and the NetBeans Integrated Development Environment. These programs make it possible to develop functional Java programs, and you also can use Sun's Java Development Kit. The kit, which is available for free on the Web at <http://java.sun.com>, is a set of command-line tools for writing, compiling, and testing Java programs.

In *Sams Teach Yourself Java 6 in 21 Days*, you are introduced to all aspects of Java software development using the most current version of the language and the best available techniques.

By the time you're finished, you'll be well acquainted with the reasons Java has become the most widely adopted programming language of the past decade.

How This Book Is Organized

Sams Teach Yourself Java 6 in 21 Days teaches you about the Java language and how to use it to create applications for any computing environment and servlets that run on web servers. By the time you have finished the book, you'll have a well-rounded knowledge of Java and the Java class libraries. Using your new skills, you will be able to develop your own programs for tasks such as web services, database connectivity, XML processing, and client/server programming.

You learn by doing in this book, creating several programs each day that demonstrate the topics being introduced. The source code for all these programs is available on the book's official website at <http://www.java21days.com>, along with other supplemental material such as answers to reader questions.

This book covers the Java language and its class libraries in 21 days, organized as three separate weeks. Each week covers a broad area of developing Java applets and applications.

In the first week, you learn about the Java language itself:

- Day 1 covers the basics—what Java is, why to learn the language, and how to create software using an innovative style of development called object-oriented programming. You create your first Java application.
- On Day 2, you dive into the fundamental Java building blocks—data types, variables, and expressions.
- Day 3 goes into detail about how to deal with objects in Java—how to create them, use their variables, call their methods, and compare them.
- On Day 4, you give Java programs cognitive skills using conditionals and work with arrays and loops.
- Day 5 fully explores the creation of classes—the basic building blocks of any Java program.
- On Day 6, you discover more about interfaces and packages, which are useful for grouping classes and organizing a class hierarchy.

- Day 7 covers three powerful features of Java—exceptions, the ability to deal with errors; threads, the ability to run parts of a program simultaneously; and assertions, a technique for making programs more reliable.

Week 2 is dedicated to the most useful classes created by Sun for use in your own Java programs:

- On Day 8, you are introduced to data structures that you can use as an alternative to strings and arrays—vectors, stacks, maps, hash tables, and bit sets—and a special for loop that makes them easier to use.
- Day 9 begins a 5-day exploration of visual programming. You learn how to create a graphical user interface using *Swing*, an extensive set of classes for interfaces, graphics, and user interactions.
- Day 10 covers more than a dozen interface components that you can use in a Java program, including buttons, text fields, sliders, scrolling text areas, and icons.
- Day 11 explains how to make a user interface look good using *layout managers*, a set of classes that determine how components on an interface are arranged.
- Day 12 concludes the coverage of Swing with *event-handling classes*, which enable a program to respond to mouse clicks and other user interactions.
- On Day 13, you learn about drawing shapes and characters on a user interface component such as an applet window.
- Day 14 demonstrates how to use Java Web Start, a technique that makes installation of a Java program as easy as clicking on a web page link, and *SwingWorker*, a class that improves application performance by using threads.

Week 3 moves into advanced topics:

- Day 15 covers input and output using *streams*, a set of classes that enable file access, network access, and other sophisticated data handling.
- Day 16 introduces object *serialization*, a way to make objects exist even when no program is running. You learn to save them to a storage medium, such as a hard disk, read them into a program, and then use them again as objects.
- On Day 17, you extend your knowledge of streams to write programs that communicate with the Internet, including socket programming, buffers, channels, and URL handling.
- Day 18 shows how to connect to relational databases using Java Database Connectivity (JDBC) and JDBC-ODBC. You learn how to exploit the capabilities of Derby, the open source database that's included for the first time in Java 6.

- Day 19 covers how to read and write RSS documents using the XML Object Model (XOM), an open source Java class library. RSS feeds, one of the most popular XML dialects in use today, enable millions of people to follow site updates and other new web content.
- Day 20 explores how to write web services clients with the language and the Apache XML-RPC class library.
- Day 21 covers two of the hottest areas in Java programming: servlets and Java Server Pages, techniques for writing Java applications that are run by web servers.

Who Should Read This Book

This book teaches the Java language to three groups:

- Novices who are relatively new to programming
- People who have been introduced to earlier versions of Java such as Java version 1.5 or 1.4
- Experienced developers in other languages, such as Visual C++, Visual Basic, or Python

When you're finished with this book, you'll be able to tackle any aspect of the Java language and be comfortable enough to tackle your own ambitious programming projects—both on and off the Web.

If you're somewhat new to programming or have never written a program before, you might wonder whether this is the right book for you. Because all the concepts in this book are illustrated with working programs, you'll be able to work your way through the subject regardless of your experience level. If you understand what variables, loops, and operators are, you'll be able to benefit from this book. You are among those who might want to read this book if any of the following rings true:

- You had some BASIC or Pascal in school, have a grasp of what programming is, and you've heard Java is easy to learn, powerful, and cool.
- You've programmed in another language for a few years, keep hearing accolades for Java, and want to see whether it lives up to its hype.
- You've heard that Java is great for web application and web services programming.

If you have never been introduced to object-oriented programming, which is the style of programming embodied by Java, don't be discouraged. This book assumes that you have no background in object-oriented design—you get a chance to learn this development methodology as you're learning Java.

If you're a complete beginner to programming, this book might move a little fast for you. Java is a good language to start with, though, and if you take it slowly and work through all the examples, you can still pick up Java and start creating your own programs.

Conventions Used in This Book

NOTE

A *Note* presents interesting, sometimes technical, pieces of information related to the surrounding discussion.

TIP

A *Tip* offers advice such as an easier way to do something.

CAUTION

A *Caution* advises you of potential problems and helps you to steer clear of disaster.

Text that you type and text that should appear on your screen is presented in monospace type:

It will look like this.

This font mimics the way text looks on your screen. Placeholders for variables and expressions appear in *monospace italic*.

The end of each lesson offers several special features: answers to commonly asked questions about that day's subject matter, a chapter-ending quiz to test your knowledge of the material, two exercises that you can try on your own, and a practice question for readers preparing for Java certification. Solutions to the exercises and the answer to the certification question can be found on the book's official website at <http://www.java21days.com>.