

C O R E
JAVA[®]

Volume II—Advanced Features

TENTH EDITION



CAY S. HORSTMANN

Core Java®

Volume II—Advanced Features

Tenth Edition

Core Java, Volume II--Advanced Features

Table of Contents

Cover

Title Page

Copyright Page

Contents

Preface

Acknowledgments

Chapter 1: The Java SE 8 Stream Library

1.1 From Iterating to Stream Operations

1.2 Stream Creation

1.3 The filter, map, and flatMap Methods

1.4 Extracting Substreams and Concatenating Streams

1.5 Other Stream Transformations

1.6 Simple Reductions

1.7 The Optional Type

1.7.1 How to Work with Optional Values

1.7.2 How Not to Work with Optional Values

1.7.3 Creating Optional Values

1.7.4 Composing Optional Value Functions with flatMap

1.8 Collecting Results

1.9 Collecting into Maps

1.10 Grouping and Partitioning

1.11 Downstream Collectors

Table of Contents

1.12 Reduction Operations

1.13 Primitive Type Streams

1.14 Parallel Streams

Chapter 2: Input and Output

2.1 Input/Output Streams

2.1.1 Reading and Writing Bytes

2.1.2 The Complete Stream Zoo

2.1.3 Combining Input/Output Stream Filters

2.2 Text Input and Output

2.2.1 How to Write Text Output

2.2.2 How to Read Text Input

2.2.3 Saving Objects in Text Format

2.2.4 Character Encodings

2.3 Reading and Writing Binary Data

2.3.1 The DataInput and DataOutput interfaces

2.3.2 Random-Access Files

2.3.3 ZIP Archives

2.4 Object Input/Output Streams and Serialization

2.4.1 Saving and Loading Serializable Objects

2.4.2 Understanding the Object Serialization File Format

2.4.3 Modifying the Default Serialization Mechanism

2.4.4 Serializing Singletons and Typesafe Enumerations

2.4.5 Versioning

2.4.6 Using Serialization for Cloning

2.5 Working with Files

2.5.1 Paths

2.5.2 Reading and Writing Files

2.5.3 Creating Files and Directories

2.5.4 Copying, Moving, and Deleting Files

Table of Contents

- 2.5.5 Getting File Information
- 2.5.6 Visiting Directory Entries
- 2.5.7 Using Directory Streams
- 2.5.8 ZIP File Systems

2.6 Memory-Mapped Files

- 2.6.1 Memory-Mapped File Performance
- 2.6.2 The Buffer Data Structure
- 2.6.3 File Locking

2.7 Regular Expressions

Chapter 3: XML

3.1 Introducing XML

- 3.1.1 The Structure of an XML Document

3.2 Parsing an XML Document

3.3 Validating XML Documents

- 3.3.1 Document Type Definitions
- 3.3.2 XML Schema
- 3.3.3 A Practical Example

3.4 Locating Information with XPath

3.5 Using Namespaces

3.6 Streaming Parsers

- 3.6.1 Using the SAX Parser
- 3.6.2 Using the StAX Parser

3.7 Generating XML Documents

- 3.7.1 Documents without Namespaces
- 3.7.2 Documents with Namespaces
- 3.7.3 Writing Documents
- 3.7.4 An Example: Generating an SVG File
- 3.7.5 Writing an XML Document with StAX

Table of Contents

3.8 XSL Transformations

Chapter 4: Networking

4.1 Connecting to a Server

4.1.1 Using Telnet

4.1.2 Connecting to a Server with Java

4.1.3 Socket Timeouts

4.1.4 Internet Addresses

4.2 Implementing Servers

4.2.1 Server Sockets

4.2.2 Serving Multiple Clients

4.2.3 Half-Close

4.3 Interruptible Sockets

4.4 Getting Web Data

4.4.1 URLs and URIs

4.4.2 Using a URLConnection to Retrieve Information

4.4.3 Posting Form Data

4.5 Sending E-Mail

Chapter 5: Database Programming

5.1 The Design of JDBC

5.1.1 JDBC Driver Types

5.1.2 Typical Uses of JDBC

5.2 The Structured Query Language

5.3 JDBC Configuration

5.3.1 Database URLs

5.3.2 Driver JAR Files

5.3.3 Starting the Database

5.3.4 Registering the Driver Class

5.3.5 Connecting to the Database

Table of Contents

5.4 Working with JDBC Statements

5.4.1 Executing SQL Statements

5.4.2 Managing Connections, Statements, and Result Sets

5.4.3 Analyzing SQL Exceptions

5.4.4 Populating a Database

5.5 Query Execution

5.5.1 Prepared Statements

5.5.2 Reading and Writing LOBs

5.5.3 SQL Escapes

5.5.4 Multiple Results

5.5.5 Retrieving Autogenerated Keys

5.6 Scrollable and Updatable Result Sets

5.6.1 Scrollable Result Sets

5.6.2 Updatable Result Sets

5.7 Row Sets

5.7.1 Constructing Row Sets

5.7.2 Cached Row Sets

5.8 Metadata

5.9 Transactions

5.9.1 Programming Transactions with JDBC

5.9.2 Save Points

5.9.3 Batch Updates

5.10 Advanced SQL Types

5.11 Connection Management in Web and Enterprise Applications

Chapter 6: The Date and Time API

6.1 The Time Line

6.2 Local Dates

6.3 Date Adjusters

Table of Contents

6.4 Local Time

6.5 Zoned Time

6.6 Formatting and Parsing

6.7 Interoperating with Legacy Code

Chapter 7: Internationalization

7.1 Locales

7.2 Number Formats

7.3 Currencies

7.4 Date and Time

7.5 Collation and Normalization

7.6 Message Formatting

7.6.1 Formatting Numbers and Dates

7.6.2 Choice Formats

7.7 Text Input and Output

7.7.1 Text Files

7.7.2 Line Endings

7.7.3 The Console

7.7.4 Log Files

7.7.5 The UTF-8 Byte Order Mark

7.7.6 Character Encoding of Source Files

7.8 Resource Bundles

7.8.1 Locating Resource Bundles

7.8.2 Property Files

7.8.3 Bundle Classes

7.9 A Complete Example

Chapter 8: Scripting, Compiling, and Annotation Processing

8.1 Scripting for the Java Platform

8.1.1 Getting a Scripting Engine

Table of Contents

- 8.1.2 Script Evaluation and Bindings
- 8.1.3 Redirecting Input and Output
- 8.1.4 Calling Scripting Functions and Methods
- 8.1.5 Compiling a Script
- 8.1.6 An Example: Scripting GUI Events
- 8.2 The Compiler API**
 - 8.2.1 Compiling the Easy Way
 - 8.2.2 Using Compilation Tasks
 - 8.2.3 An Example: Dynamic Java Code Generation
- 8.3 Using Annotations**
 - 8.3.1 An Introduction into Annotations
 - 8.3.2 An Example: Annotating Event Handlers
- 8.4 Annotation Syntax**
 - 8.4.1 Annotation Interfaces
 - 8.4.2 Annotations
 - 8.4.3 Annotating Declarations
 - 8.4.4 Annotating Type Uses
 - 8.4.5 Annotating this
- 8.5 Standard Annotations**
 - 8.5.1 Annotations for Compilation
 - 8.5.2 Annotations for Managing Resources
 - 8.5.3 Meta-Annotations
- 8.6 Source-Level Annotation Processing**
 - 8.6.1 Annotation Processors
 - 8.6.2 The Language Model API
 - 8.6.3 Using Annotations to Generate Source Code
- 8.7 Bytecode Engineering**
 - 8.7.1 Modifying Class Files
 - 8.7.2 Modifying Bytecodes at Load Time

Table of Contents

Chapter 9: Security

9.1 Class Loaders

- 9.1.1 The Class Loading Process
- 9.1.2 The Class Loader Hierarchy
- 9.1.3 Using Class Loaders as Namespaces
- 9.1.4 Writing Your Own Class Loader
- 9.1.5 Bytecode Verification

9.2 Security Managers and Permissions

- 9.2.1 Permission Checking
- 9.2.2 Java Platform Security
- 9.2.3 Security Policy Files
- 9.2.4 Custom Permissions
- 9.2.5 Implementation of a Permission Class

9.3 User Authentication

- 9.3.1 The JAAS Framework
- 9.3.2 JAAS Login Modules

9.4 Digital Signatures

- 9.4.1 Message Digests
- 9.4.2 Message Signing
- 9.4.3 Verifying a Signature
- 9.4.4 The Authentication Problem
- 9.4.5 Certificate Signing
- 9.4.6 Certificate Requests
- 9.4.7 Code Signing

9.5 Encryption

- 9.5.1 Symmetric Ciphers
- 9.5.2 Key Generation
- 9.5.3 Cipher Streams
- 9.5.4 Public Key Ciphers

Table of Contents

Chapter 10: Advanced Swing

10.1 Lists

10.1.1 The JList Component

10.1.2 List Models

10.1.3 Inserting and Removing Values

10.1.4 Rendering Values

10.2 Tables

10.2.1 A Simple Table

10.2.2 Table Models

10.2.3 Working with Rows and Columns

10.2.3.1 Column Classes

10.2.3.2 Accessing Table Columns

10.2.3.3 Resizing Columns

10.2.3.4 Resizing Rows

10.2.3.5 Selecting Rows, Columns, and Cells

10.2.3.6 Sorting Rows

10.2.3.7 Filtering Rows

10.2.3.8 Hiding and Displaying Columns

10.2.4 Cell Rendering and Editing

10.2.4.1 Rendering Cells

10.2.4.2 Rendering the Header

10.2.4.3 Editing Cells

10.2.4.4 Custom Editors

10.3 Trees

10.3.1 Simple Trees

10.3.2 Editing Trees and Tree Paths

10.3.3 Node Enumeration

10.3.4 Rendering Nodes

10.3.5 Listening to Tree Events

10.3.6 Custom Tree Models

Table of Contents

10.4 Text Components

10.4.1 Change Tracking in Text Components

10.4.2 Formatted Input Fields

10.4.2.1 Integer Input

10.4.2.2 Behavior on Loss of Focus

10.4.2.3 Filters

10.4.2.4 Verifiers

10.4.2.5 Other Standard Formatters

10.4.2.6 Custom Formatters

10.4.3 The JSpinner Component

10.4.4 Displaying HTML with the JEditorPane

10.5 Progress Indicators

10.5.1 Progress Bars

10.5.2 Progress Monitors

10.5.3 Monitoring the Progress of Input Streams

10.6 Component Organizers and Decorators

10.6.1 Split Panes

10.6.2 Tabbed Panes

10.6.3 Desktop Panes and Internal Frames

10.6.3.1 Displaying Internal Frames

10.6.3.2 Cascading and Tiling

10.6.3.3 Vetoing Property Settings

10.6.3.4 Dialogs in Internal Frames

10.6.3.5 Outline Dragging

10.6.4 Layers

Chapter 11: Advanced AWT

11.1 The Rendering Pipeline

11.2 Shapes

11.2.1 The Shape Class Hierarchy

11.2.2 Using the Shape Classes

Table of Contents

11.3 Areas

11.4 Strokes

11.5 Paint

11.6 Coordinate Transformations

11.7 Clipping

11.8 Transparency and Composition

11.9 Rendering Hints

11.10 Readers and Writers for Images

11.10.1 Obtaining Readers and Writers for Image File Types

11.10.2 Reading and Writing Files with Multiple Images

11.11 Image Manipulation

11.11.1 Constructing Raster Images

11.11.2 Filtering Images

11.12 Printing

11.12.1 Graphics Printing

11.12.2 Multiple-Page Printing

11.12.3 Print Preview

11.12.4 Print Services

11.12.5 Stream Print Services

11.12.6 Printing Attributes

11.13 The Clipboard

11.13.1 Classes and Interfaces for Data Transfer

11.13.2 Transferring Text

11.13.3 The Transferable Interface and Data Flavors

11.13.4 Building an Image Transferable

11.13.5 Transferring Java Objects via the System Clipboard

11.13.6 Using a Local Clipboard to Transfer Object References

11.14 Drag and Drop

Table of Contents

11.14.1 Data Transfer Support in Swing

11.14.2 Drag Sources

11.14.3 Drop Targets

11.15 Platform Integration

11.15.1 Splash Screens

11.15.2 Launching Desktop Applications

11.15.3 The System Tray

Chapter 12: Native Methods

12.1 Calling a C Function from a Java Program

12.2 Numeric Parameters and Return Values

12.3 String Parameters

12.4 Accessing Fields

12.4.1 Accessing Instance Fields

12.4.2 Accessing Static Fields

12.5 Encoding Signatures

12.6 Calling Java Methods

12.6.1 Instance Methods

12.6.2 Static Methods

12.6.3 Constructors

12.6.4 Alternative Method Invocations

12.7 Accessing Array Elements

12.8 Handling Errors

12.9 Using the Invocation API

12.10 A Complete Example: Accessing the Windows Registry

12.10.1 Overview of the Windows Registry

12.10.2 A Java Platform Interface for Accessing the Registry

12.10.3 Implementation of Registry Access Functions as Native Methods

Index

Table of Contents