CORE JAVA

Volume II: Advanced Features

TWELFTH EDITION

ORACLE

Cay S. Horstmann

Core Java

Volume II: Advanced Features

Twelfth Edition

Core Java, Vol. II: Advanced Features

Table of Contents

Cover
Half Title
Title Page
Copyright Page
Contents
Preface
Acknowledgments
Chapter 1: Streams
- 1.1 From Iterating to Stream Operations
- 1.2 Stream Creation
- 1.3 The filter, map, and flatMap Methods
- 1.4 Extracting Substreams and Combining Streams
- 1.5 Other Stream Transformations
- 1.6 Simple Reductions
- 1.7 The Optional Type
  - 1.7.1 Getting an Optional Value
  - 1.7.2 Consuming an Optional Value
  - 1.7.3 Pipelining Optional Values
  - 1.7.4 How Not to Work with Optional Values
  - 1.7.5 Creating Optional Values
  - 1.7.6 Composing Optional Value Functions with flatMap
  - 1.7.7 Turning an Optional into a Stream
- 1.8 Collecting Results

Table of Contents

1.9 Collecting into Maps

1.10 Grouping and Partitioning

1.11 Downstream Collectors

1.12 Reduction Operations

1.13 Primitive Type Streams

1.14 Parallel Streams

Chapter 2: InputandOutput

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

2.1.4 Text Input and Output

2.1.5 How to Write Text Output

2.1.6 How to Read Text Input

2.1.7 Saving Objects in Text Format

2.1.8 Character Encodings

2.2 Reading and Writing Binary Data

2.2.1 The DataInput and DataOutput Interfaces

2.2.2 Random-Access Files

2.2.3 ZIP Archives

2.3 Object Input/Output Streams and Serialization

2.3.1 Saving and Loading Serializable Objects

2.3.2 Understanding the Object Serialization File Format

2.3.3 Modifying the Default Serialization Mechanism

2.3.4 The readResolve and writeReplace Methods

2.3.5 Versioning

2.3.6 Using Serialization for Cloning

2.3.7 Deserialization and Security

Table of Contents

2.4 Working with Files

2.4.1 Paths

2.4.2 Reading and Writing Files

2.4.3 Creating Files and Directories

2.4.4 Copying, Moving, and Deleting Files

2.4.5 Getting File Information

2.4.6 Visiting Directory Entries

2.4.7 Using Directory Streams

2.4.8 ZIP File Systems

2.5 Memory-Mapped Files

2.5.1 Memory-Mapped File Performance

2.5.2 The Buffer Data Structure

2.6 File Locking

2.7 Regular Expressions

2.7.1 The Regular Expression Syntax

2.7.2 Matching an Entire String

2.7.3 Finding All Matches in a String

2.7.4 Groups

2.7.5 Splitting along Delimiters

2.7.6 Replacing Matches

2.7.7 Flags

Chapter 3: XML

3.1 Introducing XML

3.2 The Structure of an XML Document

3.3 Parsing an XML Document

3.4 Validating XML Documents

3.4.1 Document Type Definitions

3.4.2 XML Schema

Table of Contents

3.4.3 A Practical Example

3.5 Locating Information with XPath

3.6 Using Namespaces

3.7 Streaming Parsers

3.7.1 Using the SAX Parser

3.7.2 Using the StAX Parser

3.8 Generating XML Documents

3.8.1 Documents without Namespaces

3.8.2 Documents with Namespaces

3.8.4 Writing an XML Document with StAX

3.8.5 An Example: Generating an SVG File

3.8.3 Writing Documents

3.9 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

4.2.4 Interruptible Sockets

4.3 Getting Web Data

4.3.1 URLs and URIs

4.3.2 Using a URLConnection to Retrieve Information

4.3.3 Posting Form Data

Table of Contents

4.4 The HTTP Client

4.4.1 The HttpClient Class

4.4.2 The HttpRequest class and Body Publishers

4.4.3 The HttpResponse Interface and Body Handlers

4.4.4 Asynchronous Processing

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

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

Table of Contents

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

5.9.4 Advanced SQL Types

5.10 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

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

7.1.1 Why Locales?

7.1.2 Specifying Locales

7.1.3 The Default Locale

7.1.4 Display Names

7.2 Number Formats

7.2.1 Formatting Numeric Values

Table of Contents

7.2.2 The DecimalFormat Class

7.2.3 Currencies

7.3 Date and Time

7.4 Collation and Normalization

7.5 Message Formatting

7.5.1 Formatting Numbers and Dates

7.5.2 Choice Formats

7.6 Text Input and Output

7.6.1 Text Files

7.6.2 Line Endings

7.6.3 The Console

7.6.4 Log Files

7.6.5 The UTF-8 Byte Order Mark

7.6.6 Character Encoding of Source Files

7.7 Resource Bundles

7.7.1 Locating Resource Bundles

7.7.2 Property Files

7.7.3 Bundle Classes

7.8 A Complete Example

Chapter 8: Scripting, Compiling, and Annotation Processing

8.1 Scripting for the Java Platform

8.1.1 Getting a Scripting Engine

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

Table of Contents

8.2.1 Invoking the Compiler
8.2.2 Launching a Compilation Task
8.2.3 Capturing Diagnostics
8.2.4 Reading Source Files from Memory
8.2.5 Writing Byte Codes to Memory
8.2.6 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 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
Chapter 9: The Java Platform Module System
9.1 The Module Concept
9.2 Naming Modules
9.3 The Modular Hello, World! Program

Table of Contents

9.4 Requiring Modules
9.5 Exporting Packages
9.6 Modular JARs
9.7 Modules and Reflective Access
9.8 Automatic Modules
9.9 The Unnamed Module
9.10 Command-Line Flags for Migration
9.11 Transitive and Static Requirements
9.12 Qualified Exporting and Opening
9.13 Service Loading
9.14 Tools for Working with Modules

Chapter 10: Security

10.1 Class Loaders
10.1.1 The Class-Loading Process
10.1.2 The Class Loader Hierarchy
10.1.3 Using Class Loaders as Namespaces
10.1.4 Writing Your Own Class Loader
10.1.5 Bytecode Verification
10.2 User Authentication
10.2.1 The JAAS Framework
10.2.2 JAAS Login Modules
10.3 Digital Signatures
10.3.1 Message Digests
10.3.2 Message Signing
10.3.3 Verifying a Signature
10.3.4 The Authentication Problem
10.3.5 Certificate Signing
10.3.6 Certificate Requests

Table of Contents

10.3.7 Code Signing

10.4 Encryption

10.4.1 Symmetric Ciphers

10.4.2 Key Generation

10.4.3 Cipher Streams

10.4.4 Public Key Ciphers

Chapter 11: Advanced Swing and Graphics

11.1 Tables

11.1.1 A Simple Table

11.1.2 Table Models

11.1.3 Working with Rows and Columns

11.1.3.1 Column Classes

11.1.3.2 Accessing Table Columns

11.1.3.3 Resizing Columns

11.1.3.4 Resizing Rows

11.1.3.5 Selecting Rows, Columns, and Cells

11.1.3.6 Sorting Rows

11.1.3.7 Filtering Rows

11.1.3.8 Hiding and Displaying Columns

11.1.4 Cell Rendering and Editing

11.1.4.1 Rendering Cells

11.1.4.2 Rendering the Header

11.1.4.3 Editing Cells

11.1.4.4 Custom Editors

11.2 Trees

11.2.1 Simple Trees

11.2.1.1 Editing Trees and Tree Paths

11.2.2 Node Enumeration

11.2.3 Rendering Nodes

11.2.4 Listening to Tree Events

11.2.5 Custom Tree Models

Table of Contents

11.3 Advanced AWT

11.3.1 The Rendering Pipeline

11.3.2 Shapes

11.3.2.1 The Shape Class Hierarchy

11.3.2.2 Using the Shape Classes

11.3.3 Areas

11.3.4 Strokes

11.3.5 Paint

11.3.6 Coordinate Transformations

11.3.7 Clipping

11.3.8 Transparency and Composition

11.4 Raster Images

11.4.1 Readers and Writers for Images

11.4.1.1 Obtaining Readers and Writers for Image File Types

11.4.1.2 Reading and Writing Files with Multiple Images

11.4.2 Image Manipulation

11.4.2.1 Constructing Raster Images

11.4.2.2 Filtering Images

11.5 Printing

11.5.1 Graphics Printing

11.5.2 Multiple-Page Printing

11.5.3 Print Services

11.5.4 Stream Print Services

11.5.5 Printing Attributes

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

Table of Contents

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

12.11 Foreign Functions: A Glimpse into the Future

Index