

Contents

Acknowledgments	ix
Introduction	xxv
Chapter 1: Key Java Language Features and Libraries	1
New Language Features	1
Generics	2
Generic Types and Defining Generic Classes	3
Using Generics	5
Enhanced for Loop	7
Additions to the Java Class Library	8
Variable Arguments	9
Boxing/Unboxing Conversions	11
Unboxing Conversions	12
Valid Contexts for Boxing/Unboxing Conversions	12
Static Imports	13
Enumerations	15
Meta data	17
AnnotationDesc	20
AnnotationDesc.ElementValuePair	21
AnnotationTypeDoc	21
AnnotationTypeElementDoc	21
AnnotationValue	22
Important Java Utility Libraries	26
Java Logging	26
The Log Manager	28
The Logger Class	30
The LogRecord Class	34
The Level Class	37
The Handler Class	38
The Formatter Class	44
Stock Formatters	45
The Filter Interface	48
The ErrorManager	49
Logging Examples	49
Regular Expressions	53
The Pattern Class	58

Contents

The Matcher Class	59
The MatchResult Interface	61
Regular Expression Example	61
Java Preferences	63
The Preference Class	63
Exporting to XML	68
Using Preferences	69
Summary	71
Chapter 2: Tools and Techniques for Developing Java Solutions	73
Principles of Quality Software Development	74
Habits of Effective Software Development	75
Communicate	75
Model	75
Be Agile	75
Be Disciplined	76
Trace Your Actions to Need	76
Don't Be Afraid to Write Code	77
Think of Code as a Design, not a Product	77
Read a LOT!	78
Build Your Process from the Ground Up	78
Manage Your Configuration	78
Unit Test Your Code	79
Continuously Integrate	79
Maintaining Short Iterations	79
Measure What You Accomplished — Indirectly	80
Track Your Issues	81
Development Methodology	82
Waterfall Methodology	82
Unified Process	83
eXtreme Programming	85
Observations on Methodology	86
Practical Development Scenarios	87
Ant	87
Scenario 1	88
Scenario 2	90
Scenario 3	94
Maven	95
JUnit	98
XDoclet	101
JMeter	107
Summary	109

Chapter 3: Exploiting Patterns in Java	111
Why Patterns Are Important	112
Keys to Understanding the Java Programming Language	112
Keys to Understanding Tools Used in Java Development	113
ANT	113
JUnit	113
XDoclet	113
Keys to Developing Effective Java Solutions	113
Develop Common Design Vocabulary	114
Understand the Fundamentals of Design	114
Building Patterns with Design Principles	115
Designing a Single Class	115
Creating an Association between Classes	115
Creating an Interface	117
Creating an Inheritance Loop	117
Important Java Patterns	119
Adapter	119
The Adapter Pattern Is a Collaboration of Four Classes	120
Client	120
Adaptee	121
Adapter	121
Model-View-Controller	122
Scenario 1: Changing to the Model	123
Scenario 2: Refreshing When the Model Changes	123
Scenario 3: Initializing the Application	124
Model	124
View	125
Controller	128
Command	130
Command	130
CommandManager	131
Invoker	131
Strategy	134
Strategy	135
Context	137
Composite	138
Component	139
Leaf	139
Composite	140
Summary	142

Contents

Chapter 4: Developing Effective User Interfaces with JFC **143**

Layout Managers	144
BorderLayout	144
BoxLayout	151
FlowLayout	161
GridLayout	167
GridBagLayout	177
SpringLayout	183
CardLayout	191
JFrame and JDialog Components	197
Managing Navigation Flows in Swing Applications	214
Summary	221

Chapter 5: Persisting Your Application Using Files **223**

Application Data	224
Saving Application Data	225
A Configuration Data Model for the Imager Application	225
Java Serialization: Persisting Object Graphs	228
Key Classes	229
Serializing Your Objects	229
Configuration Example: Saving Your App's Configuration to Disk	230
Giving Your Application a Time-based License Using Serialization	235
Implementing the License	236
Implementing the Timeserver	238
Tying Your Serialization Components into the Application	239
Extending and Customizing Serialization	243
The Transient Keyword	243
Customizing the Serialization Format	243
Versioning	245
When to Use Java Serialization	247
Java Beans Long-Term Serialization: XMLEncoder/Decoder	248
Design Differences	248
XML: The Serialization Format	249
Key Classes	250
Serializing Your Java Beans	251
Robustness Demonstrated: Changing Configuration's Internal Data	252
Possible Customization	254
Persistence Delegates	255
When to Use XMLEncoder/Decoder	255

XML Schema-Based Serialization: Java API for XML Binding (JAXB)	256
Sample XML Document for Your Configuration Object	257
Defining Your XML Format with an XML Schema	259
Defining Your Data: Configuration.xsd	260
Generating JAXB Java Classes from Your Schema	263
Generated JAXB Object Graphs	265
JAXB API Key Classes	269
Marshalling and Unmarshalling XML Data	269
Creating New XML Content with JAXB-Generated Classes	270
Using JAXB-Generated Classes in Your Application	271
Implementing Your Save Action	273
Implementing Your Load Action	275
When to Use JAXB	278
Future Direction of JAXB 2.0	279
Summary	279
Chapter 6: Persisting Your Application Using Databases	281
JDBC API Overview	281
Setting Up Your Environment	283
JDBC API Usage in the Real World	283
Understanding the Two-Tier Model	283
Understanding the Three-Tier Model	284
Grasping JDBC API Concepts	285
Managing Connections	286
DriverManager Class	286
DataSource Interface	286
Understanding Statements	287
Investigating the Statement Interface	288
Exploring the PreparedStatement Interface	289
Exploring the CallableStatement Interface	292
Utilizing Batch Updates	294
Utilizing Result Sets	298
Investigating Types of Result Sets	298
Setting Concurrency of Result Sets	298
Setting Holdability of Result Sets	299
Using Result Sets	299
Examining JDBC Advanced Concepts	302
Managing Database Meta Data	302
Discovering Limitations of a Data Source	303
Determining Which Features a Data Source Supports	303
Retrieving General Information about a Data Source	304

Contents

Utilizing RowSets	308
Understanding RowSet Events	308
RowSet Standard Implementations	308
Using the New JdbcRowSetImpl	309
Connection Pooling	310
Managing Transactions	310
What Is a Transaction?	310
Standard Transactions	311
Distributed Transactions	311
Object to Relational Mapping with Hibernate	312
Exploring Hibernate's Architecture	312
Supported Database Platforms	314
Plugging Hibernate In	314
Developing with Hibernate	315
Understanding Mappings	315
Setting Hibernate Properties	317
Using Hibernate's APIs for Persistence	317
Putting It All Together: The Forum Example	320
Summary	327
<u>Chapter 7: Developing Web Applications Using the Model 1 Architecture</u>	<u>329</u>
What Is Model 1? Why Use It?	329
JSP 2.0 Overview	331
Servlet 2.4 Support	332
Expression Language Support	332
Code Reuse with *.tag and *.tagx Files	335
JSP Page Extensions (*.jspx)	336
Simple Invocation Protocol	337
Integrated Expression Language (EL)	339
JSTL 1.1 Overview	340
Function Tag Library	341
SQL Actions	342
Developing Your Web Application Visualizations with JSTL	344
Developing Your Web Application Visualizations with JSP 2.0	350
Summary	364
<u>Chapter 8: Developing Web Applications Using the Model 2 Architecture</u>	<u>365</u>
The Problem	365
What Is Model 2?	365
Why Use Model 2?	367

Developing an Application with WebWork	368
What Is Inversion of Control and Why Is It Useful?	369
Architecture	371
Interceptors	372
ValueStack	373
OGNL	373
Components	374
Extending the Framework to Support Hibernate	374
Preventing the Hanging Session	375
Defining Your Domain Model	378
Implementing Your Use Cases with Actions	384
Developing Your Views	387
Adding Contacts to the System	389
Browsing Contacts	391
Configuring Your Application	394
Adapting to Changes	397
Summary	399

Chapter 9: Interacting with C/C++ Using Java Native Interface **401**

A First Look at Java Native Interface	401
Creating the Java Code	402
Creating the Native Code and Library	403
Executing the Code	405
Java Native Interface	406
Data Types	406
Strings in JNI	406
String Example	408
Arrays in JNI	410
Array Functions	411
Array Examples	413
Working with Java Objects in C/C++	416
Accessing Fields in JNI	416
Invoking Java Methods Using JNI	419
Handling Java Exceptions in Native Code	423
Working with Object References in Native Code	425
Local References	425
Global and Weak Global References	427
Comparing References	429
Advanced Programming Using JNI	429
Java Threading	429
Native NIO Support	430

Contents

Manually Registering Native Methods	430
Reflection	432
Developing an E-Mail Client	434
System Design	434
User Interface	435
Summary	444
Chapter 10: Communicating between Java Components with RMI and EJB	445
Remote Method Invocation	445
Exploring RMI's Architecture	446
Developing RMI Applications	448
Using Threads in RMI	448
Using Dynamic Class Loading	449
Distributed Garbage Collection	449
Examining Remote Object Activations	449
TestRemoteInterface Interface	450
TestActivationImpl Class	450
TestClient Class	451
Register Class	452
Starting the Activation Tools	453
RMISChat Example	453
RMISChat Interface	454
RMISChatImpl Class	455
ChatUser Class	459
ChatApplet Class	460
Compiling the RMISChat Application	464
Enterprise JavaBeans	465
EJB Basics	465
Types of EJBs	466
Session Beans	466
Entity	466
Message Driven	466
Examining EJB Containers	467
EJB Loan Calculator Example	468
LoanObject Interface	468
LoanHome Interface	468
LoanBean Class	469
LoanClient Class	470
Examining the EJB-JAR.XML File	473
Summary	475

Chapter 11: Communicating between Java Components and Components of Other Platforms	477
Component Communication Scenarios	478
News Reader: Automated Web Browsing	478
A Bank Application: An EJB/J2EE Client	478
A Portal: Integrating Heterogeneous Data Sources and Services	478
Overview of Interprocess Communication and Basic Network Architecture	479
Sockets	480
The Java Socket API	481
Key Classes	481
Client Programming	481
Server Programming	482
Putting It All Together: An Echo Server	483
Implementing a Protocol	487
Protocol Specification	488
Proprietary Protocols and Reverse Engineering	498
Utilizing Existing Protocols and Implementations	499
Remote Method Invocation	500
Core RPC/RMI Principles	500
Marshalling and Unmarshalling	501
Protocols	503
RMI Registry	503
Distributed Objects	504
Middleware and J2EE	504
Common Object Request Broker Architecture	505
CORBA Basics	506
IDL: Interface Definition Language	507
ORB: Object Request Broker	509
Common Object Service (COS) Naming	509
IIOP: Internet InterORB Protocol	509
RMI-IIOP: Making RMI Compatible with CORBA	510
How to Turn an RMI Object into an RMI-IIOP Object	510
When to Use CORBA	512
Distributed File System Notifications: An Example CORBA System	513
The Implementation	516
Running the Example	521
Web Services	522
Evolution of the World Wide Web	523
Platform Independent RPC	526
Web Services Description Language (WSDL)	528
Simple Object Access Protocol (SOAP)	529

Contents

Weather Web Site Example	531
The Future	540
Summary	541
Chapter 12: Distributed Processing with JMS and JMX	543
Basic Concepts	544
JMS Fundamentals	544
Sending and Receiving a JMS Message	545
JMX Fundamentals	548
Using Standard MBeans	549
Deploying MBean for Management	550
Using Adaptors and Connectors	551
Building a Distributed Application	551
Deciding on the Message Type	552
Understanding the Three-Component Architecture	553
Creating a Component to Process JMS Messages	553
MessageListener	555
MessageProcessorMBean	555
JndiHelper	556
MessageProcessor	558
Processable	562
OrderProcessor	562
JMXAgent	563
Creating a Component that Directs Messages through the Business Process	564
Routeable	565
MessageRouter	565
Creating a Component to Divide Large Tasks for Parallel Processing	566
Splitable	567
MessageSplitter	567
Aggregateable	570
MessageAggregator	570
OrderAggregator	572
Deploying the Application	573
Basic Deployment	573
Advanced Deployment	578
Deploy the M-Let Service	579
Configure the Deployment Descriptor	579
Add the M-Let Configuration File to the M-Let Service	581
Summary	581

Chapter 13: Java Security	583
Java Cryptography Architecture and Java Cryptography Extension (JCA/JCE)	583
JCA Design and Architecture	584
Engine Classes	584
Calculating and Verifying Message Digests	586
Digital Signing and Verification of Data	588
Digital Key Creation and Management	592
Storing and Managing Keys	596
Algorithm Management	597
Random Number Generation	599
Certificate Management	600
Java Cryptography Extension	602
The Cipher Engine Class	603
KeyGenerator	608
SecretKeyFactory	608
Protecting Objects through Sealing	609
Computing Message Authentication Codes	611
Program Security Using JAAS	612
User Identification	612
Executing Code with Security Checks	613
Principals	614
Credentials	615
Authenticating a Subject	615
Configuration	615
LoginContext	616
Authorization	617
Summary	618
Chapter 14: Packaging and Deploying Your Java Applications	619
Examining Java CLASSPATHs	619
Investigating the Endorsed Directory	624
Exploring Java Archives	625
Manipulating JAR files	625
Examining the Basic Manifest File	628
Examining Applets and JARs	629
Signing JAR Files	630
Examining the JAR Index Option	634
Creating an Executable JAR	635

Contents

Analyzing Applets	636
Basic Anatomy of an Applet	636
Packaging an Applet for Execution	638
Examining Applet Security	639
Exploring Web Applications	639
Examining the WAR Directory Structure	640
Understanding the WAR Deployment Descriptor	640
Packaging Enterprise Java Beans	643
Inspecting Enterprise Archives	644
The EAR Descriptor File	644
Deployment Scenario	645
Jumping into Java Web Start	647
Examining the TicTacToe Example	647
Examining the TicTacToe.JNLP	648
TTTMain.java	650
TTTLogic.java	650
TTTGui.java	653
Summarizing Java Web Start	654
Using ANT with Web Archives	654
Installing ANT	654
Building Projects with ANT	655
Summary	659
References	661
Index	663
End-User License Agreement	701