

Beginning Java™ ME Platform



Ray Rischpater

Beginning Java™ ME Platform

Copyright © 2008 by Ray Rischpater

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN-13 (pbk): 978-1-4302-1061-0

ISBN-10 (pbk): 1-4302-1061-3

ISBN-13 (electronic): 978-1-4302-1062-7

ISBN-10 (electronic): 1-4302-1062-1

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Java™ and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc., in the US and other countries. Apress, Inc., is not affiliated with Sun Microsystems, Inc., and this book was written without endorsement from Sun Microsystems, Inc.

Lead Editor: Steve Anglin

Technical Reviewer: Christopher King

Editorial Board: Clay Andres, Steve Anglin, Ewan Buckingham, Tony Campbell, Gary Cornell, Jonathan Gennick, Michelle Lowman, Matthew Moodie, Jeffrey Pepper, Frank Pohlmann, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh

Project Manager: Richard Dal Porto

Copy Editor: Nicole Abramowitz

Associate Production Director: Kari Brooks-Copony

Production Editor: Katie Stence

Compositor: Patrick Cunningham

Proofreader: Liz Welch

Indexer: Brenda Miller

Artist: April Milne

Cover Designer: Kurt Krames

Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit <http://www.springeronline.com>.

For information on translations, please contact Apress directly at 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit <http://www.apress.com>.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales—eBook Licensing web page at <http://www.apress.com/info/bulksales>.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at <http://www.apress.com>.

Contents at a Glance

About the Author	xvii
About the Technical Reviewer	xix
Acknowledgments	xxi
Introduction	xxiii

PART 1 ■ ■ ■ Getting Started

■ CHAPTER 1	Mapping the Java Jungle	3
■ CHAPTER 2	Shrinking Java to Fit	19
■ CHAPTER 3	Getting Started with the NetBeans IDE	33

Intermezzo

PART 2 ■ ■ ■ CLDC Development with MIDP

■ CHAPTER 4	Introducing MIDlets	83
■ CHAPTER 5	Building User Interfaces	97
■ CHAPTER 6	Storing Data Using the Record Store	133
■ CHAPTER 7	Accessing Files and Other Data	161
■ CHAPTER 8	Using the Java Mobile Game API	193

Intermezzo

PART 3 ■ ■ ■ CDC Development

■ CHAPTER 9	Introducing Xlets and the Personal Basis Profile	223
■ CHAPTER 10	Introducing Applets and the Advanced Graphics and User Interface	253
■ CHAPTER 11	Using Remote Method Invocation	273

Intermezzo

PART 4 ■ ■ ■ Communicating with the Rest of the World

■ CHAPTER 12	Accessing Remote Data on the Network	293
■ CHAPTER 13	Accessing Web Services	331
■ CHAPTER 14	Messaging with the Wireless Messaging API	373

Intermezzo

PART 5 ■ ■ ■ Other Java ME Interfaces

■ CHAPTER 15	Securing Java ME Applications	413
■ CHAPTER 16	Rendering Multimedia Content	447
■ CHAPTER 17	Finding Your Way	499
■ CHAPTER 18	Seeking a Common Platform	523
■ APPENDIX	Finding Java APIs	539
■ INDEX	543

Contents

About the Author	xvii
About the Technical Reviewer	xix
Acknowledgments	xxi
Introduction	xxiii

PART 1 ■ ■ ■ Getting Started

■ CHAPTER 1	Mapping the Java Jungle.....	3
	Introducing the Market for Java ME.....	3
	Looking from the Device Manufacturers' Perspective.....	3
	Looking from the Operators' Perspective	4
	Looking from the Consumers' Perspective	5
	Looking Inside the Java ME Platform.....	6
	Justifying the Need for a Mobile Edition of Java	6
	Making Java Work on Mobile Devices	7
	Understanding Configurations.....	10
	Introducing the Connected Limited Device Configuration.....	10
	Introducing the Connected Device Configuration.....	12
	Understanding Profiles	12
	Introducing the Mobile Information Device Profile.....	13
	Introducing the Foundation Profile	14
	Introducing the Personal Basis Profile	14
	Introducing the Personal Profile	15
	Understanding Packages	15
	Planning Your Approach to Java ME Development	16
	Selecting Appropriate Device Targets	16
	Marketing and Selling Your Application.....	17
	Wrapping Up.....	18

CHAPTER 2	Shrinking Java to Fit	19
	Making It Fit: The CLDC	19
	Understanding the Present: CLDC 1.1	20
	Looking Back at CLDC 1.0	22
	Looking Toward the Future of the CLDC	22
	Making It Fit: The CDC	23
	Changing the Java Class Library to Fit the CLDC	24
	Changes to the java.lang Package	24
	Changes to the java.util Package	28
	Changes to the java.io Package	29
	Introducing Classes in the CLDC	30
	Changing the Java Class Library to Fit the CDC	31
	Wrapping Up	31
CHAPTER 3	Getting Started with the NetBeans IDE	33
	Selecting the NetBeans IDE	33
	Finding Your Way Around the NetBeans IDE	35
	Creating Your First CLDC/MIDP Application	37
	Walking Through the Creation of WeatherWidget	38
	Building CLDC/MIDP Applications	52
	Packaging and Executing CLDC/MIDP Applications	53
	Creating Your First CDC Application	57
	Walking Through the Creation of WeatherApplet	57
	Packaging and Executing CDC Applications	75
	Wrapping Up	77

Intermezzo

PART 2 ■ ■ ■ CLDC Development with MIDP

CHAPTER 4	Introducing MIDlets	83
	Looking at the Simplest MIDlet	83
	Understanding the MIDlet Life Cycle	85
	Packaging MIDlets	87
	Obtaining Properties and Resources	89
	Managing Startup Events and Alarms	90
	Wrapping Up	96

CHAPTER 5	Building User Interfaces	97
	Understanding the Relationship Between the Display and	
	Visible Item Objects	97
	Using Commands to Control Application Flow	101
	Introducing Basic Visible Items	104
	Introducing Items	106
	Managing Choices	112
	Introducing the Screen and Its Subclasses	114
	Collecting Visible Items Using the Form Class	114
	Alerting the User	116
	Accepting Copious Amounts of Text	119
	Showing Lists of Choices	120
	Working with the Canvas and Custom Items	122
	Controlling Drawing Behavior with a Custom Canvas	122
	Creating a Custom Item for a Screen	125
	Implementing a Custom Item	127
	Wrapping Up	131
CHAPTER 6	Storing Data Using the Record Store	133
	Peeking Inside the Record Store	133
	Using the Record Store	135
	Opening and Closing a Record Store	136
	Removing a Record Store	137
	Obtaining Information About a Record Store	137
	Accessing Records in the Record Store	138
	Adding a Record	141
	Retrieving a Record	142
	Enumerating a Record	142
	Updating a Record	144
	Removing a Record	144
	Counting Records	145
	Listening for Record Store Changes	145
	Understanding Platform Limitations of Record Stores	145
	Putting the Record Store to Work	146
	Wrapping Up	160

CHAPTER 7	Accessing Files and Other Data	161
	Introducing the FCOP	161
	Using the FCOP	163
	Determining If the FCOP Is Present	164
	Obtaining a FileConnection Instance	164
	Creating a New File or Directory	165
	Opening a File	166
	Tweaking File Attributes	166
	Deleting a File or Directory	167
	Enumerating a Directory's Contents	167
	Listening for File System Changes	168
	Putting the FCOP to Work	169
	Introducing the PIM Package	174
	Using the PIM Package	175
	Ensuring the PIM Package Is Available	176
	Opening a PIM Database	176
	Reading Records from a PIM Database	177
	Reading Fields from a PIM Record	177
	Modifying a PIM Record	182
	Adding a PIM Record	183
	Removing a PIM Entry	184
	Managing PIM Database Categories	184
	Putting the PIM Package to Work	185
	Understanding the Role Code Signing and Verification Can Play	190
	Wrapping Up	191
CHAPTER 8	Using the Java Mobile Game API	193
	Looking Inside the Mobile Game API	193
	Managing Events and Drawing	195
	Polling for Keystrokes	196
	Managing Game Execution	197
	Tying Your GameCanvas to Your MIDlet	199
	Layering Visual Elements	200
	Managing Layers	201
	Optimizing Visual Layers Using Tiling	202
	Producing Animations	205

Putting the Mobile Game API to Work	207
Implementing the Game MIDlet.	209
Implementing the Game Canvas.	210
Wrapping Up.	218

Intermezzo

PART 3 ■ ■ ■ CDC Development

■ CHAPTER 9 Introducing Xlets and the Personal Basis Profile 223

Understanding the Xlet.	223
Looking at the Xlet Life Cycle	224
Extending the Xlet Interface.	225
Using the Xlet Context.	226
Writing a Simple Xlet.	227
Looking at a Simple Xlet.	227
Understanding Xlet Dependencies	230
Developing Lightweight User Interfaces Using the PBP	233
Implementing Your Own Components for a Window Toolkit	234
Writing a Simple, Lightweight Component	236
Understanding Window Toolkit Limitations of the PBP	240
Obtaining Xlet Properties and Resources	242
Communicating with Other Xlets	243
Implementing a Shared Object	244
Sharing an Object for Other Xlets to Find	246
Using a Shared Object.	249
Wrapping Up.	251

■ CHAPTER 10 Introducing Applets and the Advanced Graphics and User Interface 253

Writing Applets for Java ME	253
Looking at the Applet Life Cycle	254
Presenting the Applet's User Interface.	256
Accessing an Applet's Context	257
Communicating Between Applets.	258

Developing User Interfaces with the AWT	260
Using AWT Containers	262
Using AWT Components	263
Handling AWT Events	264
Developing User Interfaces with the AGUI	266
Understanding Restrictions on Top-Level Windows	269
Using the AGUI's Added Input Support	269
Understanding Changes to the Drawing Algorithm	270
Wrapping Up	271

■ CHAPTER 11 Using Remote Method Invocation

Understanding Java RMI	273
Understanding the Architecture of Java RMI	274
Introducing the Java RMI Interfaces	277
Understanding the Java RMI Optional Package	278
Looking at the Requirements for the Java RMI	
Optional Package	278
Seeing What's Provided by the Java RMI	
Optional Package	279
Applying Java RMI	280
Writing the Java Interfaces for the Service	282
Implementing the Service Using Java SE	283
Generating the Stub Classes for Java SE	284
Writing the Remote Service Host Application	285
Invoking the Remote Object from the Client	286
Wrapping Up	286

Intermezzo

PART 4 ■ ■ ■ Communicating with the Rest of the World

■ CHAPTER 12	Accessing Remote Data on the Network	293
	Introducing the Generic Connection Framework	293
	Communicating with Sockets and Datagrams	300
	Using Sockets with the GCF	300
	Using Datagrams with the GCF	304
	Communicating with HTTP	306
	Reviewing HTTP	306
	Using HTTP with the GCF	309
	Putting HTTP to Work	315
	Securing Your HTTP Transaction with HTTPS	325
	Granting Permissions for Network Connections	327
	Wrapping Up	328
■ CHAPTER 13	Accessing Web Services	331
	Looking at a Web Service from the Client Perspective	331
	Considering the Architecture	333
	Exchanging Data over the Network	334
	Using XML for Data Representation	336
	Exploring XML Support for Web Services in Java ME	341
	Generating XML in Java ME Applications	343
	Introducing the J2ME Web Services Specification	355
	Introducing the kXML Parser	365
	Wrapping Up	372
■ CHAPTER 14	Messaging with the Wireless Messaging API	373
	Introducing Wireless Messaging Services	373
	Introducing Short Message Service	374
	Introducing Multimedia Messaging Service	374
	Introducing the Cell Broadcast Service	375

Introducing Wireless Messaging API	375
Creating Messages	379
Sending Messages	380
Receiving Messages	385
Managing Message Headers	385
Understanding Required Privileges When Using the WMA	386
Using the Push Registry	387
Registering Dynamically for Incoming Messages	390
Using PushRegistry APIs	390
Applying the Wireless Messaging API	391
Sending and Receiving SMS Messages	391
Sending and Receiving MMS Messages	398
Wrapping Up	407

Intermezzo

PART 5 ■ ■ ■ Other Java ME Interfaces

■ CHAPTER 15 Securing Java ME Applications	413
Understanding the Need for Security	413
Looking at Java ME's Security and Trust Services	416
Communicating with Cryptographic Hardware	
Using the APDU API	417
Communicating with Java Smart Cards Using JCRMI	420
Leveraging the SATSA High-Level APIs for Cryptography	422
Exploring the Bouncy Castle Solution to Security Challenges	425
Creating Message Digests Using the Bouncy Castle API	428
Encrypting and Decrypting Using the Bouncy Castle API	429
Creating Secure Commerce with Contactless Communications	431
Discovering Contactless Targets	432
Communicating with Contactless Targets	435
Recognizing and Generating Visual Tags	440
Wrapping Up	444

CHAPTER 16	Rendering Multimedia Content	447
	Introducing the MMAPi	448
	Understanding Basic Multimedia Concepts	448
	Understanding the Organization of the MMAPi	450
	Starting the Rendering Process	454
	Controlling the Rendering Process	458
	Capturing Media	461
	Playing Individual Tones	466
	Introducing the Java Scalable 2D Vector Graphics API	470
	Understanding Basic SVG Concepts	470
	Understanding the Organization of the SVGAPI	472
	Rendering SVG Images	474
	Modifying SVG Images	480
	Using NetBeans with SVG Images	483
	Putting the MMAPi and the SVGAPI to Work	484
	Playing Audio and Video	493
	Capturing Images	494
	Playing SVG Content	496
	Wrapping Up	497
 CHAPTER 17	 Finding Your Way	 499
	Understanding Location-Based Services	499
	Introducing the Location API	501
	Understanding the Location API	502
	Using the Location API to Determine Device Location	503
	Using the Location API to Manage Landmarks	507
	Understanding the Role That Security Plays in LBS	508
	Using the Location API	509
	Locating the User	518
	Simulating Location API Data in the Sun Java Wireless Toolkit	518
	Wrapping Up	520

■ CHAPTER 18	Seeking a Common Platform	523
	Understanding the Role JSRs Play in Fragmentation	523
	Contributing to Fragmentation and Unification	524
	Reading a JSR	525
	Dealing with Fragmentation on Your Own	527
	Understanding the JTWI	528
	Examining the JTWI Required Elements	529
	Examining the JTWI Optional Elements	529
	Understanding the MSA	530
	Understanding MSA 1.0	531
	Evolving for the Future: MSA2	534
	Wrapping Up	537
■ APPENDIX	Finding Java APIs	539
■ INDEX	543

About the Author



RAY RISCHPATER is an engineer and author with more than 15 years of experience writing about and developing for mobile-computing platforms. During this time, Ray has participated in the development of Internet technologies for Java ME, Qualcomm BREW, Palm OS, Apple Newton, and General Magic's Magic Cap, as well as several proprietary platforms. Presently, Ray is employed as the chief architect at Rocket Mobile, a wholly owned subsidiary of Buongiorno

Group. When not writing for or about mobile platforms, Ray enjoys hiking with his family and participating in public service through amateur radio in and around the San Lorenzo Valley in northern California. Ray holds a bachelor's degree in pure mathematics from the University of California, Santa Cruz and is a member of the Institute of Electrical and Electronics Engineers (IEEE), the Association for Computing Machinery (ACM), and the American Radio Relay League (ARRL). Ray's previous books include *Software Development for the QUALCOMM BREW Platform* (Apress, 2003), *Wireless Web Development, Second Edition* (Apress, 2002), and *eBay Application Development* (Apress, 2004).

About the Technical Reviewer

■ **CHRIS KING** has been writing software since childhood; today he focuses on the challenges and joys of mobile development. In recent years, he has specialized in technologies such as Java ME, Qualcomm BREW, and Android. His recent projects include messaging software that has been preloaded on millions of phones, consumer entertainment devices, middleware libraries, community organizing tools, and lifestyle applications. Chris currently serves as a lead engineer for Gravity Mobile in San Francisco.

Since moving to California, Chris has become an avid hiker, cyclist, and home cook. With any free time that remains, Chris programs for fun, writes, and devours books.

Acknowledgments

Any book today is the collaborative effort of numerous people; technical books such as this one even more so. In helping me produce this book, I owe thanks to numerous people, including some who don't realize how much they helped, and others whose names I may never know.

My son Jarod has been part of my writing career since it started; my first book and his birth nearly coincided. He is now old enough that he is writing both prose and programs on his own, giving us valuable opportunities to share in learning together. His respect for the craft of writing—shown through his asking me questions about what I am doing and how I do it—is precious to me. His ability to help me wholly forget the frustrations inherent in any large project when he and I are together is just one of the many priceless gifts he gives me.

My wife Meg embraced and encouraged this project from the beginning, despite knowing it would mean that I would spend countless hours apart from her as I researched and wrote the examples and text for this book. Her patience with my absence—extending to when I was physically present yet mumbling about some minutia of mobile-application development—bordered on the heroic at times. I cherish our relationship, and it moves me to reflect how each of us supports the other to grow and succeed.

The entire Apress staff was indispensable in bringing this book to you. Steve Anglin and Richard Dal Porto were crucial in helping start the project and shepherd it to completion. Richard was especially helpful in keeping all of the different parts of production running smoothly, even when I found myself missing the occasional deadline. Nicole Abramowitz, my copy editor, was both thorough and patient, and made innumerable improvements to this book. Katie Stence, my production editor, made the production review process painless as I saw how the book would appear in print for the first time. I also must thank those at Apress whom I have not met personally, because without their contributions, Apress would not be the successful company with which I find it so easy to work.

Chris King, this book's technical editor, is also my colleague and friend. His attention to detail frequently transcended errors of program syntax and improved my exposition of many of the concepts you encounter in this book. He fearlessly ran—and *read*—every example in this book and helped improve even the pseudocode that I use in many places as examples. I have always enjoyed working with Chris professionally, and this project cements my professional respect for him.

My colleagues at Rocket Mobile (now part of Buongiorno Group) deserve recognition not only for providing additional Java ME experience on which to draw for several examples in the book, but also for their patience and support. I must apologize to Erik Browne, Levon Dolbakian, Graham Darcy, Jonathan Jackson, and Rajiv Ramanasankaran for enduring my frequent C and Java transpositions as I wrote C code for the office during the day and Java code for the book at night. The management staff—including Young Yoon, Scott Sumner, Jim Alisago, and Wayne Yurtin—has given me the privilege of combining software engineering and writing, and has provided a climate in which both can succeed. Thank you, each and every one of you.

Many people close to me contributed additional support, whether or not they knew they were doing so. Brad Holden, Connie Rockosi, Chris Haseman, and Shane Conder are at the top of this list for giving me much-needed space, time to work, and positive encouragement during the many times when I wondered if it were possible to write a book while working full-time and having an active life outside my technical career. I am indebted to these and others for their contributions as well.

Introduction

When I set out to write this book, I was often surprised by the comments I received from friends and colleagues. Many asked me if some other platform, such as Android or the iPhone, would render Java Platform, Micro Edition (Java ME) obsolete (and non-existent, some posited) by the time the book is published. Still others pointed to the growing convergence between different lines of Java as rendering the need for separate information about Java ME obsolete. And a few remarked scathingly that the market for Java books was saturated, so investing the time to write another was an exercise in futility. You, too, may ask these questions as you decide whether or not to read this book. Perhaps you're interested in Java ME as a specific platform on which to deploy an existing product, or perhaps you're just curious as to whether you should include Java ME skills in your professional portfolio.

The Java ME platform is a highly successful one. *Billions*—yes, that's with a *b*—of devices that run Java ME are in the hands of consumers right now. Still more are on the way, including mobile phones, set-top boxes, and other devices you can't even imagine that are now in development. Java ME is deeply entrenched in the market, and yet through the Java Community Process (JCP), it evolves rapidly to address challenges raised by existing and new competing platforms, including Qualcomm BREW, Android, and the Apple iPhone.

The cross-pollination between Java ME, Java Platform, Standard Edition (Java SE), and Java Platform, Enterprise Edition (Java EE) is well recognized and will continue. Members of the JCP work carefully to introduce APIs that can be shared across these Java platforms, and many Java ME APIs are subsets of APIs proposed or developed for Java SE. In some cases, the opposite is true: Java ME APIs are being introduced into Java SE, such as the Java ME framework for communications and networking. As devices become more capable, you will see more convergence between the various Java lines, but the specific constraints on mobile devices—including ubiquitous network access, a small form factor, and scarce power, memory, and processor resources—will drive the need for specific accommodations within the Java platform. Java ME and the JCP provide a framework for vendors to make those accommodations.

There are many excellent books about Java 2 Platform, Micro Edition (J2ME)—the predecessor to Java ME—and several good books about facets of Java ME as well. However, the Java ME platform evolves and advances at a truly awe-inspiring rate, and this fact and the sheer size of Java ME make it difficult to find a good book for beginners that provides a broad foundation on which to build Java ME competency. In this book, I've worked to balance the presentation of the two profiles that comprise Java ME, because I

believe that for you to be successful, you need to understand both. At the same time, I've made explicit choices about the required and optional Java ME APIs I present, because I believe that in building this foundation, you need to understand some basic principles that arise again and again in the Java ME world, but you don't necessarily need to be able to recall from memory every method from every optional Java ME class. Given the time you have, I believe it is important for you to master the platform fundamentals, so that you're better equipped to specialize in the areas that interest you later. In short, what I *don't* present here may be as important to you as what I *do* present.

Why Should You Read This Book?

I've already partially answered this question, but it's worth recapping: Java ME is an integral part of the mobile-computing marketplace, and it's a platform that every software developer who works with mobile devices should be familiar with. Whether you need to use it daily in your job, see it as competition, or are simply curious about how it's different from the platforms for which you presently develop applications, understanding Java ME fundamentals will make you a better mobile software developer.

Whether you're new to mobile-application development or have written mobile applications for other platforms and are interested in learning what you need to know to be a Java ME developer, you should read this book. By turning equal attention to the two Java ME configurations—the Connected Limited Device Configuration (CLDC) and the Connected Device Configuration (CDC)—I prepare you to write software for either the booming mobile-phone market or the nascent market for set-top boxes and high-end mobile phones with advanced user interfaces and other capabilities. Because Java ME devices at their core are network-enabled devices, I spend a great deal of time explaining to you the APIs that Java ME uses to enable applications to communicate, and I prepare you to understand new communication schemes that Java ME may use in the coming years. Once you finish this book, you can expect to have a grasp of the most important APIs that Java ME developers use, as well as an understanding of the fundamental thinking behind the design and approach of the Java ME platform and the dynamics of the mobile-software marketplace as a whole.

However, I have some expectations of you as well. I assume you have at least some previous exposure to Java SE—both the language and some of the major classes that it supports. You may not know the difference between a `HashMap` and a `TreeMap`, but you should at least have a nodding acquaintance with Java syntax, the Java package system, and some of the basic foundation classes that you can find in the `java.lang` and `java.util` packages. Because it's an important communication tool, you should also have at least a nodding acquaintance with Unified Modeling Language (UML), as I frequently use UML class, state machine, and sequence diagrams to help illustrate the relationship between various Java ME components.

Don't worry, though, if you're new to mobile-software development. One primary aim of this book is to help you understand the dynamics of the mobile software-development marketplace, because those dynamics have and continue to influence Java ME. I firmly believe that a good software developer understands not just the platform, but the business behind the market as well. I also don't expect you to be a Java expert: you can write solid code clearly using a minimum of Java-specific language features. If I throw a closure or anonymous inner class your way, I'll let you know; my goal here is for you to learn to write mobile applications, not become the office Java guru.

In the interest of full disclosure, there may be reasons why this book isn't for you. I don't discuss every optional Java ME API in detail—for example, I omit discussions of both the Java Mobile 3D Graphics API and Java ME support for Bluetooth—because they're well covered by other texts and because they're not necessary material that every Java ME developer must know. In a similar vein, if you already have a great deal of Java ME experience under your belt, you may still learn something from this book, but your time may be better spent with a more in-depth exploration of a specific set of optional APIs that interest you. For example, another source, such as a Java Specification Request (JSR) that describes a particular API or a book on a specific topic, may be better for you. I intend this book to be a survey for beginners new to the platform that calls out the rules of the road and relevant landmarks, not an atlas of every intersection, hilltop, creek, island, and bay.

How Should You Read This Book?

Presenting Java ME to newcomers poses particular challenges, because in many ways, Java ME is really two platforms: one that's wildly successful for mobile phones, and a second that's deployed in other consumer-electronics markets. As an engineer myself, I recognize how busy you are and how you may be looking to me to give you only the information you need to solve a set of problems on a specific platform, such as a set-top box running the Java ME CDC. Consequently, I've split this book into five parts, so that you can pick and choose the information that's relevant to you.

- *Part 1, “Getting Started”*: Exposes you to the information that every Java ME developer should know: how Java ME is organized, which APIs are common across all Java ME platforms, and which tools are available. I strongly recommend you read the three chapters in this part to orient yourself to the Java ME market and mindset.
- *Part 2, “CLDC Development with MIDP”*: Explores the Java ME Connected Limited Device Configuration (CLDC) and Mobile Information Device Profile (MIDP) in detail. This configuration and associated profile comprise the most widely deployed mobile-application platform in the world, and if you're interested in writing software for mobile phones or other wireless terminals, you'll need to have a good grasp of what it offers.

- *Part 3, “CDC Development”*: Explores the Java ME Connected Device Configuration (CDC), which underpins many consumer devices today, including television set-top boxes and some advanced mobile phones. The CDC even plays a part in the Blu-ray Disc standard. The information you’ll find here is often overlooked in other introductory Java ME materials, but it plays an increasing role in Java ME development.
- *Part 4, “Communicating with the Rest of the World”*: Explains how Java ME enables the applications that you write to communicate with the rest of the Web. You’ll learn about the Generic Connection Framework (GCF)—a key addition to the Java world—as well as how Java ME enables you to work with both Internet protocols and wireless-messaging protocols.
- *Part 5, “Other Java ME Interfaces”*: Shows you a few optional APIs that every Java ME developer should know about. These interfaces are important for you to understand both because they provide capabilities nearly every application will tap (such as security and trust interfaces), and because the interfaces provide a fundamental framework that other optional Java APIs extend (such as the Mobile Media API). This part closes with a chapter examining how optional APIs fragment the Java ME platform and how the Java community works together to address this fragmentation.

A short “Intermezzo” precedes each part, helping orient you in the book. Eighteen chapters await you in the five parts:

- *Chapter 1, “Mapping the Java Jungle”*: Introduces some key vocabulary and business concepts you must understand before becoming a Java ME developer.
- *Chapter 2, “Shrinking Java to Fit”*: Describes the key transformation Java undergoes between Java SE and Java ME. If you’re a seasoned Java SE developer, you should read this chapter closely, as it tells you which language features and classes you already know that are available to you in Java ME. If you’re fairly new to Java, you should skim this chapter, but don’t be worried if you have to flip back to it occasionally.
- *Chapter 3, “Getting Started with the NetBeans IDE”*: Enables you to build your first Java ME applications using the leading software development kit (SDK) for Java ME development. You’ll learn why NetBeans is the environment of choice for developing Java ME applications, and you’ll learn how to build two simple applications from scratch using NetBeans. These sample applications are the starting points for many of the examples in subsequent chapters. Even if you decide later to switch to another SDK, this chapter will help you understand how the development tools for Java ME fit together. In the process, you’ll also get a quick overview of the major features of Java ME as you build these simple applications.

- *Chapter 4, “Introducing MIDlets”*: Begins your exploration of one of the software world’s most successful application platforms. You’ll learn about the MIDlet, which is the unit of application execution on most Java ME devices.
- *Chapter 5, “Building User Interfaces”*: Describes the hierarchy of user-interface components that are available only to Java ME developers. You’ll learn how the Java ME–provided components work and interact, as well as how to extend the Java ME component hierarchy.
- *Chapter 6, “Storing Data Using the Record Store”*: Describes the Java ME record-store model that your applications can use for persistent storage. The record store is available even on devices without a traditional file system, and it gives you the ability to store records of similar data in a searchable, persistent manner.
- *Chapter 7, “Accessing Files and Other Data”*: Provides your first exposure to an optional Java ME API—that is, an API that may not be available on all platforms. It is such an important API, however, that it’s one you should master early. You’ll need to understand how it and the record-store model presented in the previous chapter work.
- *Chapter 8, “Using the Java Mobile Game API”*: Describes the Java Mobile Game API and shows you how to write simple platform-independent games using Java ME. Game development is a complex subject; rather than get bogged down in details about game development that may not interest some readers, I emphasize the fundamentals of Java ME as they interrelate with game-development concerns.
- *Chapter 9, “Introducing Xlets and the Personal Basis Profile”*: Describes the parts of Java ME that to date have largely applied to fixed consumer electronics, such as set-top boxes. You’ll learn about the application model these devices support, as well as the interfaces they offer.
- *Chapter 10, “Introducing Applets and the Advanced Graphics and User Interface”*: Describes additional execution models available on Java ME platforms, plus support for legacy Java applets and an adaptation of Swing available on some Java ME devices.
- *Chapter 11, “Using Remote Method Invocation”*: Shows you how some Java ME devices can use Remote Method Invocation (RMI) to interact with other Java–provided services on the network.
- *Chapter 12, “Accessing Remote Data on the Network”*: Begins your foray into the communication framework supported by all Java ME devices, and shows you how to use it with Internet protocols to access data and services over the network.

- *Chapter 13, “Accessing Web Services”*: Builds on what you learn in Chapter 12 to show you how Java ME’s optional APIs and open source packages enable your applications to access web services using Extensible Markup Language (XML) and HTTP.
- *Chapter 14, “Messaging with the Wireless Messaging API”*: Shows you how to use the wireless messaging interfaces available on many Java ME devices. These interfaces enable you to send and receive messages with protocols such as Short Message Service (SMS).
- *Chapter 15, “Securing Java ME Applications”*: Looks at optional Java ME interfaces that provide extensions such as cryptography and access to smart cards, as well as interfaces that enable mobile commerce, such as the optional API for reading radio-frequency identification (RFID) cards and bar codes.
- *Chapter 16, “Rendering Multimedia Content”*: Describes Java ME’s approach to providing support for multimedia content rendering. I show you both the Mobile Media API that Java ME devices may provide, as well as an optional API for displaying and animating Scalable Vector Graphics (SVG) images.
- *Chapter 17, “Finding Your Way”*: Describes the optional Java ME interfaces that let your application determine the device location.
- *Chapter 18, “Seeking a Common Platform”*: Closes the book with a discussion of how the optional APIs that Java ME devices may provide challenge application developers like you to find sufficient devices that provide the features your applications require. I also explain how the Java community is addressing that challenge through additional device profiles such as the Java Technology for the Wireless Industry and Mobile Service Architecture (MSA).
- *Appendix, “Finding Java APIs”*: Provides you with a table of interesting mobile technologies and the JSRs that define support for those technologies. When you’re finished reading this book and want to learn more about a specific technology and how it interacts with Java ME, you can use this table to determine where to start your research.

Ideally, I’d encourage you to read all of Parts 1–4 and then whatever parts of Part 5 interest you, especially if this is your first exposure to Java ME. However, you can tackle this material in other ways as well. If you’re interested in a specific Java ME configuration, you can first read Part 1, then either Part 2 or Part 3, and then Part 4 and parts of Part 5, for example. Regardless, because some material requires you to master the material that precedes it, you should read material earlier in the book even if you skip around before you dive in to material that comes later in the book.

How Do You Get Started?

Of course, sample applications in this book are all available electronically at the Apress web site, <http://www.apress.com>. Begin by reading Chapters 1 and 2, and then download the NetBeans SDK at <http://www.netbeans.org>; if you're really in a hurry, download the SDK *now* and work through Chapter 3, so you can get a feel for what Java ME application development is all about.

I encourage you to build on what you learn here by consulting other sources; one excellent source is the Java Community Process web site at <http://www.jcp.org>, where you can find the JSRs that describe the Java ME platform (and other Java platforms and extensions to Java platforms as well). If you prefer working on the bleeding edge, the wiki for NetBeans at <http://wiki.netbeans.org> is another excellent resource, especially if you find yourself enamored with the NetBeans environment. Finally, I'll make more resources available as necessary on my web site at <http://www.lothlorien.com>.

