

# OCP

Oracle Certified Professional  
Java® SE 11 Programmer I

# STUDY GUIDE

**EXAM 1Z0-815**

Includes interactive online learning environment and study tools:

**2 custom practice exams**

**More than 250 electronic flashcards**

**Searchable key term glossary**

**JEANNE BOYARSKY  
SCOTT SELIKOFF**

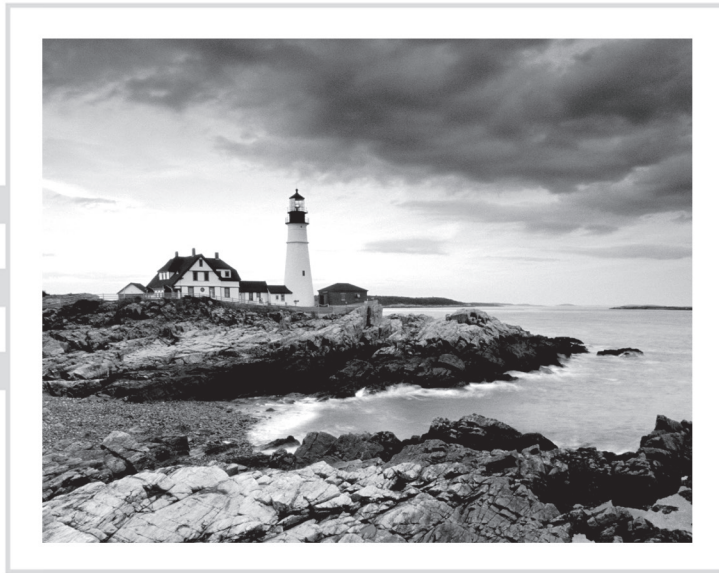
 **SYBEX®**  
A Wiley Brand



# OCP

## Oracle Certified Professional Java SE 11 Programmer I

Study Guide  
Exam 1Z0-815





# OCP

## Oracle Certified Professional Java SE 11 Programmer I

**Study Guide**

**Exam 1Z0-815**



Jeanne Boyarsky

Scott Selikoff

 **SYBEX**  
A Wiley Brand

Copyright © 2020 by John Wiley & Sons, Inc.

Published simultaneously in Canada

ISBN: 978-1-119-58470-4

ISBN: 978-1-119-58472-8 (ebk.)

ISBN: 978-1-119-58456-8 (ebk.)

Manufactured in the United States of America

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

**Limit of Liability/Disclaimer of Warranty:** The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The fact that an organization or Web site is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or Web site may provide or recommendations it may make. Further, readers should be aware that Internet Web sites listed in this work may have changed or disappeared between when this work was written and when it is read.

For general information on our other products and services or to obtain technical support, please contact our Customer Care Department within the U.S. at (877) 762-2974, outside the U.S. at (317) 572-3993 or fax (317) 572-4002.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit [www.wiley.com](http://www.wiley.com).

**Library of Congress Control Number:** 2019950141

**TRADEMARKS:** Wiley, the Wiley logo, and the Sybex logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. LPIC-1 is a registered trademark of Linux Professional Institute, Inc. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

*In memory of nine amazing years at World Maker Faire and in honor of the new FIRST robotics lab at Stuyvesant. May we have many human and robotic guests.*

*—Jeanne*

*For my youngest daughter, Elysia, you make me smile and laugh every day.*

*—Scott*





# Acknowledgments

Scott and Jeanne would like to thank numerous individuals for their contribution to this book. Thank you to Kathryn Duggan for guiding us through the process and making the book better in so many ways. Thank you to Janeice DelVecchio for being our technical editor as we wrote this book. Janeice pointed out many subtle errors in addition to the big ones. And thank you to Elena Felder for being our technical proofreader and finding the errors that we managed to sneak by Janeice. This book also wouldn't be possible without many people at Wiley, including Kenyon Brown, Pete Gaughan, Kathleen Wisor, Kim Wimpsett, Nancy Carrasco, and so many others.

Jeanne would personally like to thank Chris Kreussling for knowing almost a decade ago that she would someday write a book. He was a great mentor for many years and definitely shaped her career. Jeanne would also like to thank Cathy Sung for all the support and coaching as Jeanne grew as a tech lead. Nicolai Parlog and Juan Moreno were helpful when looking for module information. Scott was a great co-author improving everything Jeanne wrote while writing his own chapters. A big thank-you to everyone at CodeRanch.com who asked and responded to questions and comments about our Java 8 books. Finally, Jeanne would like to thank all of the new programmers at CodeRanch.com and FIRST robotics teams FRC 694, FTC 310, and FTC 479 for the constant reminders of how new programmers think.

Scott could not have reached this point without his wife Patti and family, whose love and support makes this book possible. He would like to thank his twin daughters, Olivia and Sophia, and youngest daughter, Elysia, for their patience and understanding especially when it was “time for Daddy to work in his office!” Scott would like to extend his gratitude to his wonderfully patient co-author, Jeanne, on this, their fourth book. He doesn't know how she puts up with him, but he's glad she does and thrilled at the quality of books we produce. A big thanks to Matt Dalen, who has been a great friend, sounding board, and caring father to Olivia and Adeline. Joel McNary introduced Scott to CodeRanch.com and encouraged him to post regularly, a step that changed his life. Finally, Scott would like to thank his mother and retired teacher, Barbara Selikoff, for teaching him the value of education, and his father, Mark Selikoff, for instilling in him the benefits of working hard.

Last but not least, both Scott and Jeanne would like to give a big thank-you to the readers of our Java 8 certification books. Hearing from all of you who enjoyed the book and passed the exam is a great feeling. A special thank-you to Campbell Ritchie, who fielded an almost uncountable number of the questions on CodeRanch.com, helping readers understand the material better. We'd also like to thank those who pointed out errors and made suggestions for improvements in our practice exams book. As of August 2019, the top two were Juerg Bauman and Tamas Szekeres. We would also like to thank Noorul Hameed and Nurettin Armucutu for running a tight race for third-place errata reporter.



# About the Authors

**Jeanne Boyarsky** was selected as a Java Champion in 2019. She has worked as a Java developer for more than 17 years at a bank in New York City where she develops, mentors, and conducts training. Besides being a senior moderator at CodeRanch.com in her free time, she works on the forum code base. Jeanne also mentors the programming division of a FIRST robotics team where she works with students just getting started with Java. She also speaks at several conferences each year.

Jeanne got her Bachelor of Arts degree in 2002 and her Master in Computer Information Technology degree in 2005. She enjoyed getting her Master's degree in an online program while working full-time. This was before online education was cool! Jeanne is also a Distinguished Toastmasters and a Scrum Master. You can find out more about Jeanne at [www.jeanneboyarsky.com](http://www.jeanneboyarsky.com).

**Scott Selikoff** is a professional software consultant, author, and owner of Selikoff Solutions, LLC, which provides software development solutions to businesses in the tri-state New York City area. Skilled in a plethora of software languages and platforms, Scott specializes in full-stack database-driven systems, cloud-based applications, microservice architectures, and service-oriented architectures.

A native of Toms River, New Jersey, Scott achieved his Bachelor of Arts degree from Cornell University in Mathematics and Computer Science in 2002, after three years of study. In 2003, he received his Master of Engineering degree in Computer Science, also from Cornell University.

As someone with a deep love of education, Scott has always enjoyed teaching others new concepts. He's given lectures at Cornell University and Rutgers University, as well as conferences including Oracle Code One and The Server Side Java Symposium. Scott lives in New Jersey with his loving wife, Patti, three amazing daughters, twins Olivia and Sophia and little Elysia, along with two very playful dogs, Webby and Georgette. You can find out more about Scott at [www.linkedin.com/in/selikoff](http://www.linkedin.com/in/selikoff).

Jeanne and Scott are both moderators on the CodeRanch.com forums and can be reached there for question and comments. They also co-author a technical blog called Down Home Country Coding at [www.selikoff.net](http://www.selikoff.net).

In addition to this book, Scott and Jeanne are also authors of the following best-selling Java 8 certification books: *OCA Oracle Certified Associate Java SE 8 Programmer I Study Guide* (Sybex, 2015) and *OCP Oracle Certified Professional Java SE 8 Programmer II Study Guide* (Sybex, 2016). These two books have been combined into the single release: *OCA/OCP Java SE 8 Programmer Certification Kit: Exam 1Z0-808 and Exam 1Z0-809* (Sybex 2016). They have also written a book of practice test questions for the Java 8 certification exams: *OCA/OCP Java SE 8 Programmer Practice Tests* (Sybex, 2017).



# Contents at a Glance

<i>Introduction</i>		<i>xxi</i>
<i>The Assessment Test</i>		<i>xliii</i>
<b>Chapter 1</b>	Welcome to Java	1
<b>Chapter 2</b>	Java Building Blocks	35
<b>Chapter 3</b>	Operators	79
<b>Chapter 4</b>	Making Decisions	113
<b>Chapter 5</b>	Core Java APIs	161
<b>Chapter 6</b>	Lambdas and Functional Interfaces	221
<b>Chapter 7</b>	Methods and Encapsulation	247
<b>Chapter 8</b>	Class Design	295
<b>Chapter 9</b>	Advanced Class Design	363
<b>Chapter 10</b>	Exceptions	401
<b>Chapter 11</b>	Modules	451
<b>Appendix</b>	Answers to Review Questions	491
<i>Index</i>		<i>529</i>



# Contents

<i>Introduction</i>		<i>xxi</i>
<i>The Assessment Test</i>		<i>xliii</i>
<b>Chapter 1</b>	<b>Welcome to Java</b>	<b>1</b>
	Learning About the Java Environment	2
	Major Components of Java	2
	Downloading a JDK	3
	Identifying Benefits of Java	4
	Understanding the Java Class Structure	5
	Fields and Methods	5
	Comments	6
	Classes vs. Files	8
	Writing a <i>main()</i> Method	8
	Creating a <i>main()</i> Method	9
	Passing Parameters to a Java Program	10
	Running a Program in One Line	12
	Understanding Package Declarations and Imports	13
	Wildcards	15
	Redundant Imports	15
	Naming Conflicts	16
	Creating a New Package	18
	Compiling and Running Code with Packages	19
	Using an Alternate Directory	20
	Compiling with JAR Files	22
	Creating a JAR File	23
	Running a Program in One Line with Packages	24
	Ordering Elements in a Class	24
	Code Formatting on the Exam	26
	Summary	27
	Exam Essentials	27
	Review Questions	29
<b>Chapter 2</b>	<b>Java Building Blocks</b>	<b>35</b>
	Creating Objects	36
	Calling Constructors	36
	Reading and Writing Member Fields	37
	Executing Instance Initializer Blocks	38
	Following Order of Initialization	39

Understanding Data Types	40
Using Primitive Types	40
Using Reference Types	44
Distinguishing between Primitives and Reference Types	45
Declaring Variables	46
Identifying Identifiers	46
Declaring Multiple Variables	49
Initializing Variables	50
Creating Local Variables	51
Passing Constructor and Method Parameters	52
Defining Instance and Class Variables	53
Introducing <i>var</i>	53
Managing Variable Scope	59
Limiting Scope	59
Nesting Scope	60
Tracing Scope	60
Applying Scope to Classes	61
Reviewing Scope	62
Destroying Objects	62
Understanding Garbage Collection	63
Tracing Eligibility	64
Summary	67
Exam Essentials	67
Review Questions	69
<b>Chapter 3</b>	<b>Operators</b>
	<b>79</b>
Understanding Java Operators	80
Types of Operators	80
Operator Precedence	81
Applying Unary Operators	82
Logical Complement and Negation Operators	83
Increment and Decrement Operators	84
Working with Binary Arithmetic Operators	85
Arithmetic Operators	86
Numeric Promotion	88
Assigning Values	90
Assignment Operator	90
Casting Values	90
Compound Assignment Operators	93
Assignment Operator Return Value	94
Comparing Values	95
Equality Operators	95
Relational Operators	97
Logical Operators	99
Short-Circuit Operators	100



	Making Decisions with the Ternary Operator	102
	Summary	103
	Exam Essentials	104
	Review Questions	105
<b>Chapter 4</b>	<b>Making Decisions</b>	<b>113</b>
	Creating Decision-Making Statements	114
	Statements and Blocks	114
	The <i>if</i> Statement	115
	The <i>else</i> Statement	116
	The <i>switch</i> Statement	119
	Writing <i>while</i> Loops	126
	The <i>while</i> Statement	126
	The <i>do/while</i> Statement	127
	Comparing <i>while</i> and <i>do/while</i> Loops	128
	Infinite Loops	129
	Constructing <i>for</i> Loops	130
	The <i>for</i> Loop	131
	The for-each Loop	136
	Controlling Flow with Branching	141
	Nested Loops	141
	Adding Optional Labels	142
	The <i>break</i> Statement	143
	The <i>continue</i> Statement	145
	The <i>return</i> Statement	147
	Unreachable Code	148
	Reviewing Branching	148
	Summary	149
	Exam Essentials	150
	Review Questions	151
<b>Chapter 5</b>	<b>Core Java APIs</b>	<b>161</b>
	Creating and Manipulating Strings	162
	Concatenation	163
	Immutability	164
	Important <i>String</i> Methods	165
	Method Chaining	171
	Using the <i>StringBuilder</i> Class	172
	Mutability and Chaining	173
	Creating a <i>StringBuilder</i>	174
	Important <i>StringBuilder</i> Methods	174
	Understanding Equality	177
	Comparing <i>equals()</i> and <i>==</i>	177
	The <i>String</i> Pool	179

Understanding Java Arrays	180
Creating an Array of Primitives	181
Creating an Array with Reference Variables	183
Using an Array	185
Sorting	186
Searching	187
Comparing	188
Varargs	190
Multidimensional Arrays	190
Understanding an <i>ArrayList</i>	193
Creating an <i>ArrayList</i>	193
Using an <i>ArrayList</i>	195
Wrapper Classes	199
Autoboxing and Unboxing	201
Converting Between <i>array</i> and <i>List</i>	201
Using Varargs to Create a List	203
Sorting	204
Creating Sets and Maps	204
Introducing Sets	204
Introducing Maps	205
Calculating with Math APIs	206
<i>min()</i> and <i>max()</i>	206
<i>round()</i>	207
<i>pow()</i>	207
<i>random()</i>	208
Summary	208
Exam Essentials	209
Review Questions	210
<b>Chapter 6</b>	<b>Lambdas and Functional Interfaces</b>
	<b>221</b>
Writing Simple Lambdas	222
Lambda Example	222
Lambda Syntax	225
Introducing Functional Interfaces	227
Predicate	228
Consumer	228
Supplier	229
Comparator	230
Working with Variables in Lambdas	231
Parameter List	231
Local Variables inside the Lambda Body	231
Variables Referenced from the Lambda Body	232

Calling APIs with Lambdas	234
<i>removeIf()</i>	234
<i>sort()</i>	235
<i>forEach()</i>	235
Summary	236
Exam Essentials	237
Review Questions	238
<b>Chapter 7</b>	<b>Methods and Encapsulation</b>
	<b>247</b>
Designing Methods	248
Access Modifiers	249
Optional Specifiers	250
Return Type	251
Method Name	252
Parameter List	253
Optional Exception List	253
Method Body	254
Working with Varargs	254
Applying Access Modifiers	256
Private Access	256
Default (Package-Private) Access	257
Protected Access	259
Public Access	263
Applying the <i>static</i> Keyword	264
Designing <i>static</i> Methods and Fields	264
Accessing a <i>static</i> Variable or Method	265
Static vs. Instance	266
<i>static</i> Variables	268
Static Initialization	269
Static Imports	270
Passing Data among Methods	272
Overloading Methods	275
Varargs	277
Autoboxing	277
Reference Types	277
Primitives	279
Generics	279
Arrays	280
Putting It All Together	280
Encapsulating Data	281
Summary	283
Exam Essentials	284
Review Questions	285

<b>Chapter 8</b>	<b>Class Design</b>	<b>295</b>
	Understanding Inheritance	296
	Single vs. Multiple Inheritance	297
	Inheriting <i>Object</i>	298
	Creating Classes	299
	Extending a Class	299
	Applying Class Access Modifiers	301
	Accessing the <i>this</i> Reference	302
	Calling the <i>super</i> Reference	303
	Declaring Constructors	305
	Creating a Constructor	305
	Default Constructor	306
	Calling Overloaded Constructors with <i>this()</i>	308
	Calling Parent Constructors with <i>super()</i>	310
	Constructors and <i>final</i> Fields	314
	Order of Initialization	316
	Reviewing Constructor Rules	322
	Inheriting Members	323
	Calling Inherited Members	323
	Inheriting Methods	324
	Hiding Variables	336
	Understanding Polymorphism	337
	Object vs. Reference	339
	Casting Objects	340
	The <i>instanceof</i> Operator	341
	Polymorphism and Method Overriding	342
	Overriding vs. Hiding Members	344
	Summary	346
	Exam Essentials	347
	Review Questions	349
<b>Chapter 9</b>	<b>Advanced Class Design</b>	<b>363</b>
	Creating Abstract Classes	364
	Introducing Abstract Classes	364
	Defining Abstract Methods	366
	Creating a Concrete Class	370
	Reviewing Abstract Class Rules	372
	Implementing Interfaces	373
	Defining an Interface	373
	Inserting Implicit Modifiers	377
	Inheriting an Interface	380
	Polymorphism and Interfaces	384
	Reviewing Interface Rules	386

	Introducing Inner Classes	388
	Defining a Member Inner Class	388
	Using a Member Inner Class	389
	Summary	390
	Exam Essentials	391
	Review Questions	392
<b>Chapter 10</b>	<b>Exceptions</b>	<b>401</b>
	Understanding Exceptions	402
	The Role of Exceptions	402
	Understanding Exception Types	404
	Throwing an Exception	407
	Recognizing Exception Classes	409
	<i>RuntimeException</i> Classes	409
	Checked <i>Exception</i> Classes	412
	<i>Error</i> Classes	413
	Handling Exceptions	414
	Using <i>try</i> and <i>catch</i> Statements	414
	Chaining <i>catch</i> Blocks	416
	Applying a Multi-catch Block	418
	Adding a <i>finally</i> Block	421
	Finally Closing Resources	424
	Throwing Additional Exceptions	430
	Calling Methods That Throw Exceptions	432
	Declaring and Overriding Methods with Exceptions	434
	Printing an Exception	435
	Summary	437
	Exam Essentials	438
	Review Questions	440
<b>Chapter 11</b>	<b>Modules</b>	<b>451</b>
	Introducing Modules	452
	Exploring a Module	453
	Benefits of Modules	454
	Creating and Running a Modular Program	456
	Creating the Files	457
	Compiling Our First Module	458
	Running Our First Module	460
	Packaging Our First Module	461
	Updating Our Example for Multiple Modules	463
	Updating the Feeding Module	463
	Creating a Care Module	464
	Creating the Talks Module	466
	Creating the Staff Module	469

Diving into the <i>module-info</i> File	470
<i>exports</i>	470
<i>requires transitive</i>	472
<i>provides, uses, and opens</i>	474
Discovering Modules	475
The <i>java</i> Command	475
The <i>jar</i> Command	478
The <i>jdeps</i> Command	478
The <i>jmod</i> Command	480
Reviewing Command-Line Options	481
Summary	483
Exam Essentials	484
Review Questions	485
<b>Appendix</b>	
<b>Answers to Review Questions</b>	<b>491</b>
Chapter 1: Welcome to Java	492
Chapter 2: Java Building Blocks	494
Chapter 3: Operators	498
Chapter 4: Making Decisions	501
Chapter 5: Core Java APIs	505
Chapter 6: Lambdas and Functional Interfaces	509
Chapter 7: Methods and Encapsulation	511
Chapter 8: Class Design	513
Chapter 9: Advanced Class Design	518
Chapter 10: Exceptions	522
Chapter 11: Modules	526
<i>Index</i>	529

# Introduction

This book is for those studying for the IZ0-815 (Java SE Programmer I) or IZ0-811 (Java Foundations) exam along with those looking to learn Java more deeply.

If you've taken the OCA 7 or OCA 8 exam, note that the IZ0-815 exam is a lot tougher. While covering more basic topics, the exam is at the same level of difficulty as the OCP 7 and OCP 8 exams.

In this introduction, we will cover important information about the exam before moving on to information about this book. Finally, this introduction ends with an assessment test so that you can see how much studying lays ahead of you.

## Understanding the Exam

At the end of the day, the exam is a list of questions. The more you know about the structure of the exam, the better you are likely to do. For example, knowing how many questions the exam contains allows you to manage your progress and time remaining better. In this section, we discuss the details of the exam, along with some history of previous certification exams.

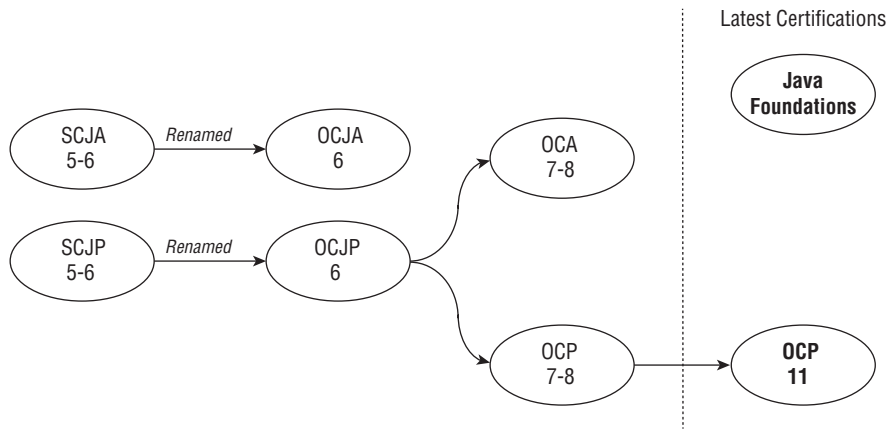
## Choosing Which Exam to Take

Java is about 25 years old, celebrating being “born” in 1995. As with anything 25 years old, there is a good amount of history and variation between different versions of Java. Over the years, the certification exams have changed to cover different topics. The names of the exams have even changed.

Oracle released two exams each for Java 7 and Java 8. The first exam tended to be easier and completing it granted you the title of Oracle Certified Associate (OCA). The second exam was a lot more difficult, with much longer questions, and completing it granted you the title of Oracle Certified Professional (OCP).

Oracle did not release an exam for Java 9 or Java 10, probably because neither of these are Long Term Support (LTS) releases (more on that in Chapter 1, “Welcome to Java”). With Java 11, Oracle decided to discontinue both the OCA certification and its associated exam. You still have to take two exams to earn an OCP title. There's also a basic Java Foundations exam that we will describe shortly.

Figure I.1 shows these past and current Java certifications. This image is helpful if you run into material online that references older exams. It is also helpful if you have an older certification and are trying to determine where it fits in.

**FIGURE I.1** Past and current Java certifications

The first exam, and the one this book is designed to prepare you for, is the 1Z0-815 Programmer I exam. It covers a lot of basic Java properties including class structure, methods, inheritance, and modules. It is somewhat similar, but significantly more difficult, than the previous OCA 7/8 exams. If you've taken a previous OCA exam, it might help to think of the 1Z0-815 exam as containing much of the same OCA content, but with the level of difficulty of the original OCP exam. This means questions are longer, answers often have multiple parts, and the level of depth of the material is significantly more difficult.

The second exam is the 1Z0-816 Programmer II exam. It is quite similar to the previous OCP 7/8 exams, with a number of new topics such as annotations, modules, and security added in. We've included notes throughout this book on some topics that aren't in scope for the 1Z0-815 exam but will be when you study for the 1Z0-816 exam.



Oracle has also released a 1Z0-817 OCP Upgrade Exam for those who hold an existing Sun Certified Programmer 6 (SCJP 6), OCP 6, OCP 7, or OCP 8 title. The objectives for the upgrade exam are quite similar to the 1Z0-816 Programmer II exam.

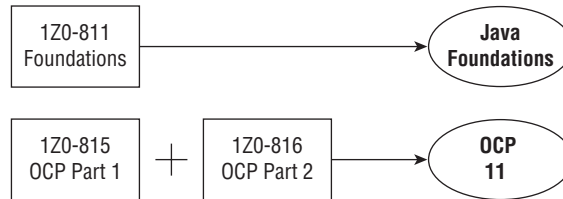
Oracle also offers a 1Z0-811 Java Foundations exam. This is often for a novice programmer or nonprogrammers. If you are planning to take the Java Foundations exam, this book prepares you as well. You will get to skip some parts of the book, so be sure to read the objectives/chapter mapping later in this introduction. The Java Foundations exam is an easier exam but does not serve as a prerequisite for the OCP certification. If you are considering taking the Java Foundations exam, please see the linked blog post from our book page to weigh the pros and cons of each exam:

<http://www.selikoff.net/ocp11-1>



Figure I.2 reviews the exams you need to take in order to earn the latest Java certifications. This book will prepare you for your choice of the 1Z0-815 and the 1Z0-811 exams.

**FIGURE I.2** Latest Java certification exams



### Broad Objectives

In previous certification exams, the list of exam objectives tended to include specific topics, classes, and APIs that you needed to know for the exam. For example, take a look at a previous objective for the 1Z0-808 (OCA 8) exam:

*Compare and contrast the features and components of Java such as: platform independence, object orientation, encapsulation, etc.*

Now compare it with the equivalent objective for the 1Z0-815 exam:

*Identify key features of the Java language*

Notice the difference? The older version is more detailed and describes specific topics you will need to understand. The newer version is a lot vaguer. It also gives the exam writers a lot more freedom to, say, insert a new feature without having to update the list of objectives.

So, how do you know what to study? By reading this study guide, of course! We've spent years studying the certification exams, in all of their forms, and have carefully cultivated topics, material, and practice questions that we are confident can lead to successfully passing the exam.

## Changes to the Exam

At the time of this book being published, all three OCP 11 certification exams contain 80 questions and have a duration of 3 hours. The 1Z0-815 and 1Z0-816 exams require a passing score of 63%, while the 1Z0-817 exam requires a passing score of 61%. The 1Z0-811 Java Foundation exam is a little different than the OCP exams. It contains 75 questions and has a duration of 2.5 hours, and it requires a score of 65% to pass.

Oracle has a tendency to fiddle with the length of the exam and the passing score once it comes out. Oracle also likes to “tweak” the exam topics over time. It wouldn’t be a surprise for Oracle to make minor changes to the exam objectives, the number of questions, or the passing score after this book goes to print.

If there are any changes to the exam after this book is published, we will note them on the book page of our blog:

<http://www.selikoff.net/ocp11-1>

## Exam Questions

The 1Z0-815 exam consists of multiple-choice questions. There are between four and seven possible answers. If a question has more than one answer, the question specifically states exactly how many correct answers there are. This book does not do that. We say “Choose all that apply” to make the questions harder. This means the questions in this book are generally harder than those on the exam. The idea is to give you more practice so you can spot the correct answer more easily on the real exam.

If you read about older versions of the exam online, you might see references to drag-and-drop questions. These questions had you do a puzzle on how to complete a piece of code. Luckily, these are no longer on the exam.

Many of the questions on the exam are code snippets rather than full classes. Saving space by not including imports leaves room for lots of other code. In Chapter 1, we provide advice on reading code in various formats.

## Out-of-Scope Material

When you take the exam, you may see some questions that appear to be out of scope. *Don’t panic!* Oftentimes, these questions do not require knowing anything about the topic to answer the question. For example, after reading this book you should be able to spot that the following does not compile, even if you have no idea what `LocalDate` and `ChronoUnit` are:

```
final LocalDate holiday = LocalDate.now();  
holiday = LocalDate.now().plus(5,ChronoUnit.HOURS);
```

While the classes and enums used in this question are not in scope for the exam, the reason it does not compile is. In particular, you should know that you cannot reassign a variable marked `final`.

See, not so scary is it? Expect to see at least a few structures on the exam that you are not familiar with. If they aren’t part of your exam preparation material, then you don’t need to understand them to answer the question.

## Question Topic Tips

The following list of topics is meant to give you an idea of the types of questions and oddities that you might come across on the exam. Being aware of these categories of such questions will help you get a higher score on the exam.

**Questions with Extra Information Provided** Imagine the question includes a statement that `XMLParseException` is a checked exception. It's fine if you don't know what an `XMLParseException` is or what XML is for that matter. (If you are wondering, it is a format for data.) This question is a gift. You know the question is about checked and unchecked exceptions.

**Questions with Embedded Questions** To answer some questions on the exam, you may have to actually answer two or three subquestions. For example, the question may contain two blank lines, and the question may ask you to choose the two answers that fill in each blank. In some cases, the two answer choices are not related, which means you're really answering multiple questions, not just one! Another place this is seen is in a question that includes a `print()` statement that outputs multiple variables. Each question has to be tracked independently. These questions are among the most difficult and time-consuming on the exam because they contain multiple, often independent, questions to answer. Unfortunately, the exam does not give partial credit, so take care when answering questions like these.

**Questions with Unfamiliar APIs** Suppose a question shows a method with a `Path` parameter. If you see a class or method that wasn't covered in this book, assume it works as you would expect. Some of these APIs you might come across, such as `LocalDate`, were on the Java 8 exam and are not part of the Java 11 exams. Assume that the part of the code using that API is correct and look very hard for other errors.

**Questions with Unfamiliar Concepts** You might see some more advanced Java concepts like an enum or inner classes that use aspects not covered in this book. While you need to know more for the 1Z0-816 exam, the 1Z0-815 only tests the basics of these features. Again, you can assume the unfamiliar provided code is correct and the question is testing something else.

**Questions with Made Up or Incorrect Concepts** In the context of a word problem, the exam may bring up a term or concept that does not make any sense such as saying an interface inherits from a class, which is not a correct statement. In other case, they may use a keyword that does not exist in Java, like `struct`. For these, you just have to read them carefully and recognize when the exam is using invalid terminology.

**Questions That Are Really Out of Scope** When introducing new questions, Oracle includes them as unscored questions at first. This allows them to see how real exam takers do without impacting your score. You will still receive the number of questions as the exam lists. However, a few of them may not count. These unscored questions may contain out-of-scope material or even errors. They will not be marked as unscored, so you still have to do your best to answer them. Follow the previous advice to assume that anything you haven't seen before is correct. That will cover you if the question is being counted!

# Reading This Book

It might help to have some idea about how this book has been written. This section contains details about some of the common structures and features you will find in this book, where to go for additional help, and how to obtain bonus material for this book.

## Who Should Buy This Book

If you want to become certified as a Java programmer, this book is definitely for you. If you want to acquire a solid foundation in Java and your goal is to prepare for the exam, this book is also for you. You'll find clear explanations of the concepts you need to grasp and plenty of help to achieve the high level of professional competency you need in order to succeed in your chosen field.

This book is intended to be understandable to anyone who has a tiny bit of Java knowledge. If you've never read a Java book before, we recommend starting with a book that teaches programming from the beginning and then returning to this study guide.

This book is for anyone from high school students to those beginning their programming journey to experienced professionals who need a review for the certification.

## How This Book Is Organized

This book consists of 11 chapters plus supplementary information: an Appendix, a glossary (online), this introduction, and the bonus exam. You might have noticed that there are more than 11 exam objectives. We organized what you need to know to make it easy to learn and remember. Each chapter begins with a list of the objectives that are covered in that chapter.

The chapters are organized as follows:

**Chapter 1: Welcome to Java** describes the basics of Java such as how to run a program. It also includes the benefits of Java and key terminology.

**Chapter 2: Java Building Blocks** focuses on variables such as primitives and object data types and scoping variables. It also discusses garbage collection.

**Chapter 3: Operators** explains operations with variables. It also talks about casting and the precedence of operators.

**Chapter 4: Making Decisions** covers on core logical constructs such as conditionals and loops.

**Chapter 5: Core Java APIs** introduces you to `String`, `StringBuilder`, array, and various types.

**Chapter 6: Lambdas and Functional Interfaces** shows how use lambdas and four key functional interfaces. The focus is implementing and calling `Predicate`, `Consumer`, `Supplier`, and `Comparator`.

**Chapter 7: Methods and Encapsulation** explains how to write methods. It also shows the four access modifiers.

**Chapter 8: Class Design** covers constructors and superclasses. It also includes method overriding.

**Chapter 9: Advanced Class Design** adds interfaces and abstract classes. It also introduces inner classes.

**Chapter 10: Exceptions** shows the different types of exception classes and how to use them. It also includes different uses of try statements.

**Chapter 11: Modules** details the benefits of the new module feature. It shows how to compile and run module programs from the command line.

At the end of each chapter, you'll find a few elements you can use to prepare for the exam.

**Summary** This section reviews the most important topics that were covered in the chapter and serves as a good review.

**Exam Essentials** This section summarizes highlights that were covered in the chapter. You should be able to convey the information requested.

**Review Questions** Each chapter concludes with at least 20 review questions. You should answer these questions and check your answers against the ones provided in the Appendix. If you can't answer at least 80% of these questions correctly, go back and review the chapter, or at least those sections that seem to be giving you difficulty.



The review questions, assessment test, and other testing elements included in this book are *not* derived from the real exam questions, so don't memorize the answers to these questions and assume that doing so will enable you to pass the exam. You should learn the underlying topic, as described in the text of the book. This will let you answer the questions provided with this book *and* pass the exam. Learning the underlying topic is also the approach that will serve you best in the workplace—the ultimate goal of a certification.

To get the most out of this book, you should read each chapter from start to finish before going to the chapter-end elements. They are most useful for checking and reinforcing your understanding. Even if you're already familiar with a topic, you should skim the chapter. There are a number of subtleties to Java that you could easily not encounter even when working with Java for years.

## Conventions Used in This Book

This book uses certain typographic styles to help you quickly identify important information and to avoid confusion over the meaning of words such as on-screen prompts. In particular, look for the following styles:

- *Italicized text* indicates key terms that are described at length for the first time in a chapter. (Italics are also used for emphasis.)

- A monospaced font indicates code or command-line text.
- *Italicized monospaced text* indicates a variable.

In addition to these text conventions, which can apply to individual words or entire paragraphs, a few conventions highlight segments of text.



A note indicates information that's useful or interesting. It is often something to pay special attention to for the exam.

### Sidebars

A sidebar is like a note but longer. The information in a sidebar is useful, but it doesn't fit into the main flow of the text.



### Real World Scenario

#### Real-World Scenario

A real-world scenario is a type of sidebar that describes a task or an example that's particularly grounded in the real world. This is something that is useful in the real world but is not going to show up on the exam.

## Getting Help

Both of the authors are moderators at [CodeRanch.com](http://CodeRanch.com). This site is a quite large and active programming forum that is very friendly toward Java beginners. It has a forum just for this exam called Programmer Certification. It also has a forum called Beginning Java for non-exam-specific questions. As you read the book, feel free to ask your questions in either of those forums. It could be you are having trouble compiling a class or that you are just plain confused about something. You'll get an answer from a knowledgeable Java programmer. It might even be one of us.

## Interactive Online Learning Environment and Test Bank

We've put together some really great online tools to help you pass the IZ0-815 exam. The interactive online learning environment that accompanies this study guide provides a test bank and study tools to help you prepare for the exam. By using these tools you can dramatically increase your chances of passing the exam on your first try.