OCA Java SE 8 Programmer I Exam Guide (Exams 1Z0-808)

by Kathy Sierra and Bert Bates

Introduction

ORGANIZATION

This book is organized in such a way as to serve as an in-depth review for the OCA 8 exam, for both experienced Java professionals and those in the early stages of experience with Java technologies. Each chapter covers at least one major aspect of the exam, with an emphasis on the "why" as well as the "how to" of programming in the Java language.

Practice exam software with two 80-question exams is available for download.

WHAT THIS BOOK IS NOT

You will not find a beginner's guide to learning Java in this book. All 500+ pages of this book are dedicated solely to helping you pass the exam. If you are brand new to Java, we suggest you spend a little time learning the basics. You should not start with this book until you know how to write, compile, and run simple Java programs. We do not, however, assume any level of prior knowledge of the individual topics covered. In other words, for any given topic (driven exclusively by the actual exam objectives), we start with the assumption that you are new to that topic. So we assume you're new to the individual topics, but we assume that you are not new to Java.

We also do not pretend to be both preparing you for the exam and simultaneously making you a complete Java being. This is a certification exam study guide, and it's very clear about its mission. That's not to say that preparing for the exam won't help you become a better Java programmer! On the contrary, even the most experienced Java developers often claim that having to prepare for the certification exam made them far more knowledgeable and well-rounded programmers than they would have been without the exam-driven studying.

DIGITAL CONTENT

For more information about the practice exam software, please see the Appendix.

SOME POINTERS

Once you've finished reading this book, set aside some time to do a thorough review. You might want to return to the book several times and make use of all the methods it offers for reviewing the material:

1. *Re-read all the Two-Minute Drills, or have someone quiz you*. You also can use the drills as a way to do a quick cram before the exam. You might want to make some flash cards out of 3 × 5 index cards that have the Two-Minute Drill material on them.

2. *Re-read all the Exam Watch notes*. Remember that these notes are written by authors who helped create the exam. They know what you should expect—and what you should be on the lookout for.

3. *Re-take the Self Tests*. Taking the tests right after you've read the chapter is a good idea because the questions help reinforce what you've just learned. However, it's an even better idea to go back later and do all the questions in the book in one sitting. Pretend that you're taking the live exam. (Whenever you take the Self Tests, mark your answers on a separate piece of paper. That way, you can run through the questions as many times as you need to until you feel comfortable with the material.)

4. *Complete the exercises*. The exercises are designed to cover exam topics, and there's no better way to get to know this material than by practicing. Be sure you understand why you are performing each step in each exercise. If there is something you are not clear on, re-read that section in the chapter.

5. *Write lots of Java code*. We'll repeat this advice several times. When we wrote this book, we wrote hundreds of small Java programs to help us do our research. We have heard from hundreds of candidates who have passed the exam, and in almost every case, the candidates who scored extremely well on the exam wrote lots of code during their studies. Experiment with the code samples in the book, create horrendous lists of compiler errors—put away your IDE, crank up the command line, and write code!

INTRODUCTION TO THE MATERIAL IN THE BOOK

The OCA 8 exam is considered one of the hardest in the IT industry, and we can tell you from experience that a large chunk of exam candidates goes in to the test unprepared. As programmers, we tend to learn only what we need to complete our current project, given the insane deadlines we're usually under.

But this exam attempts to prove your complete understanding of the Java language, not just the parts of it you've become familiar with in your work.

Experience alone will rarely get you through this exam with a passing mark, because even the things you think you know might work just a little differently than you imagined. It isn't enough to be able to get your code to work correctly; you must understand the core fundamentals in a deep way, and with enough breadth to cover virtually anything that could crop up in the course of using the language.

Who Cares About Certification?

Employers do. Headhunters do. Programmers do. Passing this exam proves three important things to a current or prospective employer: you're smart; you know how to study and prepare for a challenging test; and, most of all, you know the Java language. If an employer has a

choice between a candidate who has passed the exam and one who hasn't, the employer knows that the certified programmer does not have to take time to learn the Java language.

But does it mean that you can actually develop software in Java? Not necessarily, but it's a good head start. To really demonstrate your ability to develop (as opposed to just your knowledge of the language), you should consider pursuing the Java Developer Exam, where you're given an assignment to build a program, start to finish, and submit it for an assessor to evaluate and score.

TAKING THE PROGRAMMER'S EXAM

In a perfect world, you would be assessed for your true knowledge of a subject, not simply how you respond to a series of test questions. But life isn't perfect, and it just isn't practical to evaluate everyone's knowledge on a one-to-one basis.

For most of its certifications, Oracle evaluates candidates using a computer-based testing service operated by Pearson VUE. To discourage simple memorization, Oracle exams present a potentially different set of questions to different candidates. In the development of the exam, hundreds of questions are compiled and refined using beta testers. From this large collection, questions are pulled together from each objective and assembled into many different versions of the exam.

Each Oracle exam has a specific number of questions, and the test's duration is designed to be generous. The time remaining is always displayed in the corner of the testing screen. If time expires during an exam, the test terminates, and incomplete answers are counted as incorrect.

Exam Watch

Many experienced test-takers do not go back and change answers unless they have a good reason to do so. Only change an answer when you feel you may have misread or misinterpreted the question the first time. Nervousness may make you second-guess every answer and talk yourself out of a correct one.

After completing the exam, you will receive an e-mail from Oracle telling you that your results are available on the Web. As of winter 2017, your results can be found at certview.oracle.com. If you want a printed copy of your certificate, you must make a specific request.

Question Format

Oracle's Java exams pose questions in multiple-choice format.

Multiple-Choice Questions

In earlier versions of the exam, when you encountered a multiple-choice question, you were not told how many answers were correct; but with each version of the exam, the questions have become more difficult, so today, each multiple-choice question tells you how many answers to choose. The Self Test questions at the end of each chapter closely match the format, wording, and difficulty of the real exam questions, with two exceptions:

- Whenever we can, our questions will *not* tell you how many correct answers exist (we will say "Choose all that apply"). We do this to help you master the material. Some savvy test-takers can eliminate wrong answers when the number of correct answers is known. It's also possible, if you know how many answers are correct, to choose the most plausible answers. Our job is to toughen you up for the real exam!
- The real exam typically numbers lines of code in a question. Sometimes we do not number lines of code—mostly so that we have the space to add comments at key places. On the real exam, when a code listing starts with line 1, it means that you're looking at an entire source file. If a code listing starts at a line number greater than 1, that means you're looking at a partial source file. When looking at a partial source file, assume the code you can't see is correct. (For instance, unless explicitly stated, you can assume that a partial source file will have the correct import and package statements.)

Exam Watch

When you find yourself stumped answering multiple-choice questions, use your scratch paper (or whiteboard) to write down the two or three answers you consider the strongest, then underline the answer you feel is most likely correct. Here is an example of what your scratch paper might look like when you've gone through the test once:

- 21. B or C
- 33. A or C

This is extremely helpful when you mark the question and continue on. You can then return to the question and immediately pick up your thought process where you left off. Use this technique to avoid having to re-read and rethink questions. You will also need to use your scratch paper during complex, text-based scenario questions to create visual images to better understand the question. This technique is especially helpful if you are a visual learner.

Tips on Taking the Exam

The number of questions and passing percentages for every exam are subject to change. Always check with Oracle before taking the exam, at www.Oracle.com.

You are allowed to answer questions in any order, and you can go back and check your answers after you've gone through the test. There are no penalties for wrong answers, so it's better to at least attempt an answer than to not give one at all.

A good strategy for taking the exam is to go through once and answer all the questions that come to you quickly. You can then go back and do the others. Answering one question might jog your memory for how to answer a previous one.

Be very careful on the code examples. Check for syntax errors first: count curly braces, semicolons, and parentheses, and then make sure there are as many left ones as right ones. Look for capitalization errors and other such syntax problems before trying to figure out what the code does.

Many of the questions on the exam will hinge on subtleties of syntax. You will need to have a thorough knowledge of the Java language in order to succeed.

This brings us to another issue that some candidates have reported. The testing center is supposed to provide you with sufficient writing implements so you can work problems out "on paper." In some cases, the centers have provided inadequate markers and dry-erase boards that are too small and cumbersome to use effectively. We recommend that you call ahead and verify that you will be supplied with a sufficiently large whiteboard, sufficiently fine-tipped markers, and a good eraser. What we'd really like to encourage is for everyone to complain to Oracle and Pearson VUE and have them provide actual pencils and at least several sheets of blank paper.

Tips on Studying for the Exam

First and foremost, give yourself plenty of time to study. Java is a complex programming language, and you can't expect to cram what you need to know into a single study session. It is a field best learned over time, by studying a subject and then applying your knowledge. Build yourself a study schedule and stick to it, but be reasonable about the pressure you put on yourself, especially if you're studying in addition to your regular duties at work.

One easy technique to use in studying for certification exams is the 15-minutes-per-day effort. Simply study for a minimum of 15 minutes every day. It is a small but significant commitment. If you have a day where you just can't focus, then give up at 15 minutes. If you have a day where it flows completely for you, study longer. As long as you have more of the "flow days," your chances of succeeding are excellent.

We strongly recommend you use flash cards when preparing for the programmer's exams. A flash card is simply a 3 × 5 or 4 × 6 index card with a question on the front and the answer on the back. You construct these cards yourself as you go through a chapter, capturing any topic you think might need more memorization or practice time. You can drill yourself with them by reading the question, thinking through the answer, and then turning the card over to see if you're correct. Or you can get another person to help you by holding up the card with the question facing you and then verifying your answer. Most of our students have found these to be tremendously helpful, especially because they're so portable that while you're in study mode, you can take them everywhere. Best not to use them while driving, though, except at red lights. We've taken ours everywhere—the doctor's office, restaurants, theaters, you name it.

Certification study groups are another excellent resource, and you won't find a larger or more willing community than on the Coderanch.com Big Moose Saloon certification forums. If you have a question from this book, or any other mock exam question you may have stumbled upon, posting a question in a certification forum will get you an answer in nearly all cases within a day—usually, within a few hours.

Finally, we recommend that you write a lot of little Java programs! During the course of writing this book, we wrote hundreds of small programs, and if you listen to what the most successful candidates say (you know, those guys who got 98 percent), they almost always report that they wrote a lot of code.

SCHEDULING YOUR EXAM

You can purchase your exam voucher from either Oracle or Pearson VUE. Visit Oracle.com (follow the training/certification links) or visit PearsonVue.com for exam scheduling details and locations of test centers.

Arriving at the Exam

As with any test, you'll be tempted to cram the night before. Resist that temptation. You should know the material by this point, and if you're groggy in the morning, you won't remember what you studied anyway. Get a good night's sleep.

Arrive early for your exam; it gives you time to relax and review key facts. Take the opportunity to review your notes. If you get burned out on studying, you can usually start your exam a few minutes early. We don't recommend arriving late. Your test could be cancelled, or you might not have enough time to complete the exam.

When you arrive at the testing center, you'll need to provide current, valid photo identification. Visit PearsonVue.com for details on the ID requirements. They just want to be sure that you don't send your brilliant Java guru next-door neighbor who you've paid to take the exam for you.

Aside from a brain full of facts, you don't need to bring anything else to the exam room. In fact, your brain is about all you're allowed to take into the exam!

All the tests are closed book, meaning you don't get to bring any reference materials with you. You're also not allowed to take any notes out of

the exam room. The test administrator will provide you with a small marker board. If you're allowed to, we do recommend that you bring a water bottle or a juice bottle (call ahead for details of what's allowed). These exams are long and hard, and your brain functions much better when it's well hydrated. In terms of hydration, the ideal approach is to take frequent, small sips. You should also verify how many "bio-breaks" you'll be allowed to take during the exam!

Leave your phone in the car. It will only add stress to the situation, since they are not allowed in the exam room, and can sometimes still be heard if they ring outside of the room. Purses, books, and other materials must be left with the administrator before entering the exam.

Once in the testing room, you'll be briefed on the exam software. You might be asked to complete a survey. The time you spend on the survey is *not* deducted from your actual test time—nor do you get more time if you fill out the survey quickly. Also, remember the questions you get on the exam will *not* change depending on how you answer the survey questions. Once you're done with the survey, the real clock starts ticking and the fun begins.

The testing software allows you to move forward and backward between questions. Most important, there is a Mark check box on the screen—this will prove to be a critical tool, as explained in the next section.

TEST-TAKING TECHNIQUES

Without a plan of attack, candidates can become overwhelmed by the exam or become sidetracked and run out of time. For the most part, if you are comfortable with the material, the allotted time is more than enough to complete the exam. The trick is to keep the time from slipping away during any one particular problem.

Your obvious goal is to answer the questions correctly and quickly, but other factors can distract you. Here are some tips for taking the exam more efficiently.

Size Up the Challenge

First, take a quick pass through all the questions in the exam. "Cherry-pick" the easy questions, answering them on the spot. Briefly read each question, noticing the type of question and the subject. As a guideline, try to spend less than 25 percent of your testing time in this pass.

This step lets you assess the scope and complexity of the exam, and it helps you determine how to pace your time. It also gives you an idea of where to find potential answers to some of the questions. Sometimes the wording of one question might lend clues or jog your thoughts for another question.

If you're not entirely confident in your answer to a question, answer it anyway, but check the Mark box to flag it for later review. In the event you run out of time, at least you've provided a "first guess" answer, rather than leaving it blank.

Second, go back through the entire test, using the insight you gained from the first go-through. For example, if the entire test looks difficult, you'll know better than to spend more than a minute or two on each question. Create a pacing with small milestones—for example, "I need to answer 10 questions every 15 minutes."

At this stage, it's probably a good idea to skip past the time-consuming questions, marking them for the next pass. Try to finish this phase before you're 50 to 60 percent through the testing time.

Third, go back through all the questions you marked for review, using the Review Marked button in the question review screen. This step includes taking a second look at all the questions you were unsure of in previous passes, as well as tackling the time-consuming ones you deferred until now. Chisel away at this group of questions until you've answered them all.

If you're more comfortable with a previously marked question, unmark the Review Marked button now. Otherwise, leave it marked. Work your way through the time-consuming questions now, especially those requiring manual calculations. Unmark them when you're satisfied with the answer.

By the end of this step, you've answered every question in the test, despite having reservations about some of your answers. If you run out of time in the next step, at least you won't lose points for lack of an answer. You're in great shape if you still have 10 to 20 percent of your time remaining.

Review Your Answers

Now you're cruising! You've answered all the questions, and you're ready to do a quality check. Take yet another pass (yes, one more) through the entire test, briefly re-reading each question and your answer.

Carefully look over the questions again to check for "trick" questions. Be particularly wary of those that include a choice of "Does not compile." Be alert for last-minute clues. You're pretty familiar with nearly every question at this point, and you may find a few clues that you missed before.

The Grand Finale

When you're confident with all your answers, finish the exam by submitting it for grading. After you finish your exam, you'll receive an e-mail from Oracle giving you a link to a page where your exam results will be available. As of this writing, you must ask for a hard copy certificate specifically or one will not be sent to you.

Retesting

If you don't pass the exam, don't be discouraged. Try to have a good attitude about the experience, and get ready to try again. Consider yourself a little more educated. You'll know the format of the test a little better, and you'll have a good idea of the difficulty level of the questions you'll get next time around.

If you bounce back quickly, you'll probably remember several of the questions you might have missed. This will help you focus your study efforts in the right area.

Ultimately, remember that Oracle certifications are valuable because they're hard to get. After all, if anyone could get one, what value would it have? In the end, it takes a good attitude and a lot of studying, but you can do it!

OBJECTIVES MAP

The following table describes the exam objectives and where you will find coverage for them in the book. (Note: We have summarized some of the descriptions you will find on the Oracle.com website.)

Oracle Certified Associate Java SE 8 Programmer (Exam 1Z0-808)

Exam Objective	Study Guide Coverage
Java Basics	
Define the scope of variables (1.1)	Chapter 3
Define the structure of a Java class (1.2)	Chapters 1 and 2
Create executable Java applications with a main method (1.3)	Chapter 1
Import other Java packages to make them accessible in your code (1.4)	Chapter 1
Compare and contrast features of Java (1.5)	Chapter 1
Working with Java Data Types	
Declare and initialize variables (2.1)	Chapters 1 and 3
Differentiate between object reference variables and primitive variables (2.2)	Chapters 1, 2, and 3
Know how to read or write to object fields (2.3)	Whole book
Explain an object's lifecycle (creation, "dereference," and garbage collection) (2.4)	Chapter 3
Use wrapper classes such as Boolean, Double, Integer (2.5)	Chapter 6
Using Operators and Decision Constructs	
Use Java operators (3.1)	Chapters 1 and 4
Test equality between Strings and other objects using == and equals() (3.2)	Chapters 1 and 4
Create if and if/else and ternary constructs (3.4)	Chapters 4 and 5
Use a switch statement (3.5)	Chapter 5
Creating and Using Arrays	
Declare, instantiate, initialize and use a one-dimensional array (4.1)	Chapters 3 and 6
Declare, instantiate, initialize and use multi-dimensional arrays (4.2)	Chapters 3 and 6
Using Loop Constructs	
Create and use while loops (5.1)	Chapter 5
Create and use for loops including the enhanced for loop (5.2)	Chapter 5
Create and use do/while loops (5.3)	Chapter 5
Compare loop constructs (5.4)	Chapter 5
Use break and continue (5.5)	Chapter 5
Working with Methods and Encapsulation	
Create methods with arguments and return values (6.1)	Chapter 2
Apply the static keyword to methods and fields (6.2)	Chapters 1 and 2
Create and overload constructors (6.3)	Chapters 1 and 2
Apply access modifiers (6.4)	Chapter 1
Apply encapsulation principles to a class (6.5)	Chapters 2 and 6
Determine the effect upon object references and primitive values when they are passed into methods that change the values (6.6)	Chapter 3
Working with Inheritance	
Describe inheritance and its benefits (7.1)	Chapter 2

Develop code that demonstrates the use of polymorphism (7.2)	Chapter 2
Determine when casting is necessary (7.3)	Chapter 2
Use super and this to access objects and constructors (7.4)	Chapter 2
Use abstract classes and interfaces (7.5)	Chapters 1 and 2
Handling Exceptions	
Differentiate among checked exceptions, RuntimeExceptions, and Errors (8.1)	Chapter 5
Create a try-catch block and determine how exceptions alter normal program flow (8.2)	Chapter 5
Describe the advantages of Exception handling (8.3)	Chapter 5
Create and invoke a method that throws an exception (8.4)	Chapter 5
Recognize common exception classes (8.5)	Chapter 5
Working with Selected Classes from the Java API	
Manipulate data using the StringBuilder class (9.1)	Chapter 6
Create and manipulate Strings (9.2)	Chapter 6
Create and manipulate calendar data (9.3)	Chapter 6
Declare and use an ArrayList (9.4)	Chapter 6
Write a simple Lambda expression that consumes a Lambda Predicate expression (9.5)	Chapter 6