

**CompuScholar, Inc.**  
Alignment to  
"Oracle Certified Associate (OCA), Java SE 8 Programmer 1 "  
Certification Exam Requirements

**Oracle Exam Details:**

<b>Exam Title:</b>	Oracle Certified Associate, Java SE 8 Programmer 1
<b>Exam Code(s):</b>	1Z0-808
<b>Exam Link:</b>	<a href="#">Java SE 8 Programmer 1</a>

**CompuScholar Course Details:**

<b>Course Title:</b>	Java Programming (Abridged)
<b>Course ISBN:</b>	978-0-9887070-4-7
<b>Course Year:</b>	2018

OR

<b>Course Title:</b>	Java Programming (AP)
<b>Course ISBN:</b>	978-0-9887070-2-3
<b>Course Year:</b>	2018

**Note 1:** Citation(s) listed may represent a subset of the instances where objectives are met throughout the course.

**Note 2:** Citation(s) for a "Lesson" refer to the "Lesson Text" elements and associated "Activities" within the course, unless otherwise noted. The "Instructional Video" components are supplements designed to introduce or re-enforce the main lesson concepts, and the Lesson Text contains full details.

**Course Description**

The Oracle Certified Associate, Java SE 8 Programmer 1 exam is designed to test basic knowledge of the Java programming language and object-oriented programming skills. Preparation for this exam can generally be accomplished within a single school year.

EITHER the CompuScholar "Java Programming (Abridged)" OR the "Java Programming" courses can be used to prepare for this exam. Nearly all exam topics are covered, with a few minor exceptions noted as "n/a" below.

## Exam Requirements

Java Basics	CITATION(S)
Define the scope of variables	Chapter 10, Lesson 2
Define the structure of a Java class	Chapter 10, Lesson 2
Create executable Java applications with a main method; run a Java program from the command line; produce console output	Chapter 2, Lesson 2 and many chapter activities
Import other Java packages to make them accessible in your code	Chapter 2, Lesson 4
Compare and contrast the features and components of Java such as: platform independence, object orientation, encapsulation, etc.	Chapter 2, Lesson 1 Chapters 10, 11, 15

Working With Java Data Types	CITATION(S)
Declare and initialize variables (including casting of primitive data types)	Chapter 4, Lesson 2 Chapter 5, Lesson 1
Differentiate between object reference variables and primitive variables	Chapter 4, Lesson 2 Chapter 5, Lesson 1
Know how to read or write to object fields	Chapter 10, Lesson 3
Explain an Object's Lifecycle (creation, "dereference" and garbage collection)	n/a
Develop code that uses wrapper classes such as Boolean, Double, and Integer	Chapter 4, Lesson 2 Chapter 5, Lesson 5

Using Operators and Decision Constructs	CITATION(S)
Use Java operators; use parentheses to override operator precedence	Chapter 7, Lesson 1
Test equality between Strings and other objects using == and equals ()	Chapter 5, Lesson 2 Chapter 15, Lesson 5
Create if and if/else and ternary constructs	Chapter 7, Lesson 2
Use a switch statement	Chapter 7, Lesson 3

<b>Creating and Using Arrays</b>	<b>CITATION(S)</b>
Declare, instantiate, initialize and use a one-dimensional array	Chapter 14, Lesson 1
Declare, instantiate, initialize and use multi-dimensional array	Chapter 14, Lesson 1

<b>Using Loop Constructs</b>	<b>CITATION(S)</b>
Create and use while loops	Chapter 7, Lesson 5
Create and use for loops including the enhanced for loop	Chapter 7, Lesson 4 Chapter 14, Lesson 3
Create and use do/while loops	Chapter 7, Lesson 5
Compare loop constructs	Chapter 7, Lesson 4 Chapter 7, Lesson 5
Use break and continue	Chapter 7, Lesson 3 Chapter 7, Lesson 4

<b>Working with Methods and Encapsulation</b>	<b>CITATION(S)</b>
Create methods with arguments and return values; including overloaded methods	Chapter 8, Lesson 2
Apply the static keyword to methods and fields	Chapter 11, Lesson 3
Create and overload constructors; differentiate between default and user defined constructors	Chapter 11, Lesson 1 Chapter 15, Lesson 6
Apply access modifiers	Chapter 10, Lesson 3
Apply encapsulation principles to a class	Chapter 10, Lesson 1 Chapter 10, Lesson 2 Chapter 10, Lesson 3
Determine the effect upon object references and primitive values when they are passed into methods that change the values	Chapter 8, Lesson 3

<b>Working with Inheritance</b>	<b>CITATION(S)</b>
Describe inheritance and its benefits	Chapter 15 (all Lessons) Chapter 16 (all Lessons)

Develop code that makes use of polymorphism; develop code that overrides methods; differentiate between the type of a reference and the type of an object	Chapter 15 (all Lessons) Chapter 16 (all Lessons)
Determine when casting is necessary	Chapter 15, Lesson 3
Use super and this to access objects and constructors	Chapter 15, Lesson 6
Use abstract classes and interfaces	Chapter 11, Lesson 2 Chapter 15, Lesson 2

Handling Exceptions	CITATION(S)
Differentiate among checked exceptions, RuntimeExceptions and Errors	Chapter 9, Lesson 1 (exceptions described but not differentiated between checked/runtime)
Create a try-catch block and determine how exceptions alter normal program flow	Chapter 9, Lesson 2
Describe the advantages of Exception handling	Chapter 9, Lesson 1 Chapter 9, Lesson 2
Create and invoke a method that throws an exception	Chapter 9 Activity Chapter 18, Lesson 2
Recognize common exception classes (such as NullPointerException, ArithmeticException, ArrayIndexOutOfBoundsException, ClassCastException)	Chapter 9, Lesson 1

Working with Selected classes from the Java API	CITATION(S)
Manipulate data using the StringBuilder class and its methods	n/a
Create and manipulate Strings	Chapter 5 (all lessons)
Create and manipulate calendar data using classes from java.time.LocalDateTime, java.time.LocalDate, java.time.LocalTime, java.time.format.DateTimeFormatter, java.time.Period	n/a
Declare and use an ArrayList of a given type	Chapter 14, Lesson 1
Write a simple Lambda expression that consumes a Lambda Predicate expression	n/a