

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

**Oracle Exam Details:**

<b>Exam Title:</b>	Oracle Certified Associate, Java SE 8 Programmer I
<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
<b>Course ISBN:</b>	978-1-946113-99-3
<b>Course Year:</b>	2023

**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 reinforce the main lesson concepts, and the Lesson Text contains full details.

### Course Description

The Java SE 8 Oracle Certified Associate (OCA) certification helps you build a foundational understanding of Java, and gaining this certification credential is the first of two steps in demonstrating you have the high-level skills needed to become a professional Java developer.

## Exam Requirements

Java Basics	CITATION(S)
Define the scope of variables	Chapter 15, Lesson 2
Define the structure of a Java class	Chapter 14
Create executable Java applications with a main method; run a Java program from the command line; produce console output	Chapter 2, Lesson 3 Chapter 3, Lesson 3 Chapter 28
Import other Java packages to make them accessible in your code	Chapter 5, Lessons 1, 6 Chapter 14, Lesson 2
Compare and contrast the features and components of Java such as: platform independence, object orientation, encapsulation, etc.	Chapter 2, Lesson 2 Chapters 14, 15

Working With Java Data Types	CITATION(S)
Declare and initialize variables (including casting of primitive data types)	Chapter 3, Lesson 2 Chapter 4, Lesson 3
Differentiate between object reference variables and primitive variables	Chapter 5, Lesson 2
Know how to read or write to object fields	Chapter 5, Lesson 4
Explain an Object's Lifecycle (creation, "dereference" and garbage collection)	Chapter 5, Lesson 4 (creation only)
Develop code that uses wrapper classes such as Boolean, Double, and Integer	Chapter 7, Lessons 1,2

Using Operators and Decision Constructs	CITATION(S)
Use Java operators; use parentheses to override operator precedence	Chapter 8, Lesson 2
Test equality between Strings and other objects using == and equals ()	Chapter 6, Lesson 1 Chapter 9, Lesson 1 Chapter 23, Lesson 4
Create if and if/else and ternary constructs	Chapter 8, Lessons 2, 3
Use a switch statement	Chapter 8, Lesson 4

<b>Creating and Using Arrays</b>	<b>CITATION(S)</b>
Declare, instantiate, initialize and use a one-dimensional array	Chapter 18
Declare, instantiate, initialize and use multi-dimensional array	Chapter 21

<b>Using Loop Constructs</b>	<b>CITATION(S)</b>
Create and use while loops	Chapter 12, Lesson 2
Create and use for loops including the enhanced for loop	Chapter 12, Lesson 1 Chapter 18, Lesson 3
Create and use do/while loops	Chapter 12, Lesson 2
Compare loop constructs	Chapter 12, Lesson 2
Use break and continue	Chapter 12, Lesson 3

<b>Working with Methods and Encapsulation</b>	<b>CITATION(S)</b>
Create methods with arguments and return values; including overloaded methods	Chapter 14, Lesson 5 Chapter 15, Lesson 4
Apply the static keyword to methods and fields	Chapter 5, Lesson 5 Chapter 16
Create and overload constructors; differentiate between default and user defined constructors	Chapter 5, Lesson 5 Chapter 14, Lesson 4
Apply access modifiers	Chapter 14
Apply encapsulation principles to a class	Chapter 15, Lesson 3
Determine the effect upon object references and primitive values when they are passed into methods that change the values	Chapter 14, Lesson 5

<b>Working with Inheritance</b>	<b>CITATION(S)</b>
Describe inheritance and its benefits	Chapter 22
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 23

Determine when casting is necessary	Chapter 22, Lesson 3
Use super and this to access objects and constructors	Chapter 23, Lesson 3
Use abstract classes and interfaces	Chapter 15, Lesson 5 Chapter 23, Lesson 2

Handling Exceptions	CITATION(S)
Differentiate among checked exceptions, unchecked exceptions, and Errors	Chapter 10, Lesson 1 (exceptions described but not differentiated between checked/unchecked)
Create a try-catch block and determine how exceptions alter normal program flow	Chapter 10, Lesson 2
Describe the advantages of Exception handling	Chapter 10, Lessons 1, 2
Create and invoke a method that throws an exception	Chapter 10, Lesson 3 Chapter 25, Lesson 2
Recognize common exception classes (such as NullPointerException, ArithmeticException, ArrayIndexOutOfBoundsException, ClassCastException)	Chapter 10, 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, Lesson 2 Chapter 6
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 19
Write a simple Lambda expression that consumes a Lambda Predicate expression	n/a