Oracle Certified Associate (OCA)

Java SE 7 Programmer I

Exam 1Z0-803

1. Basics
   1. Variable Scope
   2. Structure of a Java Class
   3. Create executable applications with main method
   4. Import other Java packages
2. Java Data Types
   1. Declare and initialize variables
   2. Differentiate between object reference variables and primitive variables
   3. Read or write object fields
   4. Explain an object's life cycle
   5. Call methods on objects
   6. Manipulate data using the String-Builder class and its methods
3. Using operators and decision constructs
   1. Use Java operators
   2. Use parenthesis to override operator precedence
   3. Test equality between strings and other objects using == and equals()
   4. Create if and if-else constructs
   5. Use a switch statement
4. Creating and using arrays
   1. Declare, instantiate, initialize, and use a one-dimensional array
   2. Declare, instantiate, initialize, and use a multi-dimensional array
   3. Declare and use an ArrayList
5. Using loop constructs
   1. Create and use while loops
   2. Create and use for loops, including the enhance for loop
   3. Create and use do-while loops
   4. Compare loop constructs
   5. Use break and continue
6. Working with methods and encapsulation
   1. Creating methods with arguments and return values
   2. Apply the static keyword to methods and fields
   3. Create an overloaded method
   4. Differentiate between default and user-defined constructs
   5. Create and overload constructors
   6. Apply access modifiers
   7. Apply encapsulation principles to a class
   8. Determine the effect upon object references and primitive values when they are passed into methods that change the values
7. Working with inheritance
   1. Implement inheritance
   2. Develop code that demonstrates the use of polymorphism
   3. Differentiate when casting is necessary
   4. Use super and this to access objects and constructors
   5. Use abstract classes and interfaces
8. Handling exceptions
   1. Differentiate among checked exceptions, Runtime Exceptions, and Errors
   2. Create a try-catch block and determine how exceptions alter normal program flow
   3. Describe what exceptions are used for in Java
   4. Invoke a method that throws an exception
   5. Recognize common exception classes and categories