8

Classes and Objects: A Deeper Look, Solutions

Instead of this absurd division into sexes, they ought to class people as static and dynamic.
—Evelyn Waugh

Is it a world to hide virtues in?
—William Shakespeare

But what, to serve our private ends, Forbids the cheating of our friends?
—Charles Churchill

This above all: to thine own self be true.
—William Shakespeare

Don't be "consistent," but be simply true.

Oliver Wendell Holmes, Jr.

Objectives

In this chapter you'll learn:

- Encapsulation and data hiding.

- To use keyword `this`.

- To use `static` variables and methods.

- To import `static` members of a class.

- To use the `enum` type to create sets of constants with unique identifiers.

- To declare `enum` constants with parameters.

- To organize classes in packages to promote reuse.

Self-Review Exercises

8.1   Fill in the blanks in each of the following statements:

a) When compiling a class in a package, the javac command-line option _____ specifies where to store the package and causes the compiler to create the package's directories if they do not exist.

ANS: `-d`.

b) `String` class `static` method _____ is similar to method `System.out.printf`, but returns a formatted `String` rather than displaying a `String` in a command window.

ANS: `format`.

c) If a method contains a local variable with the same name as one of its class's fields, the local variable _____ the field in that method's scope.

ANS: shadows.

d) The _____ method is called by the garbage collector just before it reclaims an object's memory.

ANS: `finalize`.

e) A(n) _____ declaration specifies one class to import.

ANS: single-type-import.

f) If a class declares constructors, the compiler will not create a(n) _____.

ANS: default constructor.

g) An object's _____ method is called implicitly when an object appears in code where a `String` is needed.

ANS: `toString`.

h) *Get* methods are commonly called _____ or _____.

ANS: accessor methods, query methods.

i) A(n) _____ method tests whether a condition is true or false.

ANS: predicate.

j) For every `enum`, the compiler generates a `static` method called _____ that returns an array of the `enum`'s constants in the order in which they were declared.

ANS: `values`.

k) Composition is sometimes referred to as a(n) _____ relationship.

ANS: *has-a*.

l) A(n) _____ declaration contains a comma-separated list of constants.

ANS: `enum`.

m) A(n) _____ variable represents classwide information that is shared by all the objects of the class.

ANS: `static`.

n) A(n) _____ declaration imports one `static` member.

ANS: single `static` import.

o) The _____ states that code should be granted only the amount of privilege and access that it needs to accomplish its designated task.

ANS: principle of least privilege.

p) Keyword _____ specifies that a variable is not modifiable.

ANS: `final`.

q) There can be only one _____ in a Java source-code file, and it must precede all other declarations and statements in the file.

ANS: `package` declaration.

r) A(n) _____ declaration imports only the classes that the program uses from a particular package.

ANS: type-import-on-demand.

s) The compiler uses a(n) _____ to locate the classes it needs in the classpath.
ANS: class loader.

t) The classpath for the compiler and JVM can be specified with the _____ option to the `javac` or `java` command, or by setting the _____ environment variable.
ANS: `-classpath`, `CLASSPATH`.

u) *Set* methods are commonly called _____ because they typically change a value.
ANS: mutator methods.

v) A(n) _____ imports all `static` members of a class.
ANS: `static` import on demand.

w) The `public` methods of a class are also known as the class's _____ or _____.
ANS: `public` services, `public` interface.

Exercises

*NOTE: Solutions to the programming exercises are located in the **ch08solutions** folder. Each exercise has its own folder named **ex08_##** where ## is a two-digit number representing the exercise number. For example, exercise 8.4's solution is located in the folder **ex08_04**.*

8.2 Explain the notion of package access in Java. Explain the negative aspects of package access.
ANS: Package access allows a class, method, or variable to be accessible within the same package. Package access does not promote good OOP when applied to an instance variable because it destroys the notion of information hiding.

8.3 What happens when a return type, even `void`, is specified for a constructor?
ANS: It is treated as a method and is not considered to be a constructor.