Chapter 8 Critical Thinking Questions

1 Explain the difference between a has-a and is-a relationship among classes.

The Has-a relationship is where a class contains a class and does not derive things such as Bank has Account and has-a Customer.

Classes that are derived from existing classes use a is-a relationship example Circle goes to Disk goes to Puck.

2 If a base class has a public method go() and a derived class has a public method stop(), which method would be available to an object of the derived class.

Derived class gets the method from its superclass, the object of the derived class is its superclass and base class is a synonym of superclass. The available methods would be stop() and go().

3 Compare and contrast implementing an abstract method to overriding a method.

An abstract method is declared with the word abstract and contains a method declaration but no body. A subclass is only declared with abstract when its superclass is declared with abstract. A subclass rewriting a superclass method is overriding the superclass method. A subclass contains body code.

4 Compare and contrast an abstract class to an interface.

An interface is a class with method declarations that have no implementations. An interface can’t be inherited like abstract classes can be. Both an interface and an abstract class can be implemented in a class and add behavior to a class. An interface class and an abstract class does not provide a hierarchy for the class.

6 Use the following classes to answer the questions below:

A What type of method is doThat() in Wo? A public integer.

B What is Wo? An interface class.

C Why is doThat() implemented in Roo? Roo implements Wo so Roo has to implement Wo’s methods.

D List the methods available to a Roo object?

doThis(), doNow(), doThat().

E How does the implementation of doThis() in Roo affect the implementation of doThis() in Bo?

The implementation of doThis() in Roo does not affect the implementation of doThis() in Bo.

F What action does the statement super(1) in Roo perform?

Roo extends Bo and the super sets the Bo z integer to 1.

G Can the doThis() method in Bo be called from a Roo object? If so, how?

A Roo object is its superclass also known as Bo, so no doThis() in Bo can't be called in Bo again.

H Can a method in Roo call the doThis() method in Bo? If so, how?

Yes, the same way the example class did, rewriting the method.