

Assignment 02

2/22/2010

1. Answer the following questions.
(From H.M. Deitel and P.J. Deitel, Java How to Program)
 - (a) What is the relationship between a client of an object and the object's public members?
 - (b) What if an object has all of its attributes declared as private? What can you do?
 - (c) What is the purpose of keyword new? Explain what happens when this keyword is used in an application.
 - (d) What is a default constructor? How are an object's instance variables initialized if a class has only a default constructor?
2. Each of the Java files represents a complete source file. Determine whether each of these files will compile. If they won't compile, how would you fix them, and if they do compile, what would be their output? (Fig. 1)
(From K. Sierra and B. Bates, Head First Java)
3. Exercise 1 - Complex Numbers (package name: p01)
 - (a) Create a class for managing complex numbers. Complex numbers have the form $realPart + imaginaryPart * i$ where $i = \sqrt{-1}$
Use double variables to represent the private data of the class. Provide a constructor function that enables an object of this class to be initialized when it is declared. The constructor should contain default values in case no initializers are provided. Provide public member functions for each of the following:
 - i. Accessor methods
 - ii. Printing Complex numbers in the form $a + bi$ where a is the real part and b is the imaginary part.
 - (b) Create a class called ComplexOperations for performing arithmetic with complex numbers. Provide public member functions for each of the following:
 - i. Addition of two Complex numbers: The real parts are added together and the imaginary parts are added together.
 - ii. Subtraction of two Complex numbers: The real part of the right operand is subtracted from the real part of the left operand and the imaginary part of the right operand is subtracted from the imaginary part of the left operand.
 - (c) Create a MainClass to serve as a wrapper for a main method that instantiates two Complex objects and performs the operations.
4. Exercise 2 - Dog Class (package name: p02)

<p style="text-align: center;">A</p> <pre> class TapeDeck { boolean canRecord = false; void playTape() { System.out.println("tape playing"); } void recordTape() { System.out.println("tape recording"); } } class TapeDeckTestDrive { public static void main(String [] args) { t.canRecord = true; t.playTape(); if (t.canRecord == true) { t.recordTape(); } } } </pre>	<p style="text-align: center;">B</p> <pre> class DVDPlayer { boolean canRecord = false; void recordDVD() { System.out.println("DVD recording"); } } class DVDPlayerTestDrive { public static void main(String [] args) { DVDPlayer d = new DVDPlayer(); d.canRecord = true; d.playDVD(); if (d.canRecord == true) { d.recordDVD(); } } } </pre>
---	---

Figure 1: Java code analysis.

- (a) Create a class that represents a Dog object, use whatever attributes you think might be relevant.
- (b) Define at least three methods, being bark() one of them. This method will produce a different kind of “bark sound” depending on the size of the dog. (Sinde a small dog’s bark is different from a big dog’s bark)