**Assignment 3**

1. Explain polymorphism.

Polymorphism is the ability of a programming language to present the same interface for several different underlying data types. Polymorphism is the ability of different objects to respond in a unique way to the same message.

1. What is overloading?

Overloading in Java is the ability to define more than one method with the same name in a class. The compiler is able to distinguish between the methods because of their method signatures.

1. What is overriding?

In any object-oriented programming language, Overriding is a feature that allows a subclass or child class to provide a specific implementation of a method that is already provided by one of its super-classes or parent classes.

1. What does the final mean in this method: public void doSomething(**final** Car aCar){}

A final variable can be explicitly initialized only once. A reference variable declared final can never be reassigned to refer to a different object. However, the data within the object can be changed. So, the state of the object can be changed but not the reference. With variables, the final modifier often is used with static to make the constant a class variable.

1. Suppose in question 4, the Car class has a method setColor(Color color){…}, inside doSomething method, Can we call aCar.setColor(red);?

Yes we can.

1. Can we declare a static variable inside a method?

No we can’t.

1. What is the difference between interface and abstract class?

Main difference is methods of a Java interface are implicitly abstract and cannot have implementations. A Java abstract class can have instance methods that implements a default behavior. Variables declared in a Java interface is by default final. An abstract class may contain non-final variables.

1. Can an abstract class be defined without any abstract methods?

yes, you can declare abstract class without defining an abstract method in it.

1. Since there is no way to create an object of abstract class, what’s the point of constructors of abstract class?

The main purpose of the constructor is to initialize the newly created object. In abstract class, we have an instance variable, abstract methods, and non-abstract methods.

1. What is a native method?

Native methods are Java methods that start in a language other than Java. Native methods can access system-specific functions and APIs that are not available directly in Java. The use of native methods limits the portability of an application, because it involves system-specific code.

1. What is marker interface?

A marker interface is an interface that has no methods or constants inside it. It provides run-time type information about objects, so the compiler and JVM have additional information about the object. A marker interface is also called a tagging interface.

1. Why to override equals and hashCode methods?

You must override hashCode() in every class that overrides equals(). Failure to do so will result in a violation of the general contract for Object.hashCode(), which will prevent your class from functioning properly in conjunction with all hash-based collections, including HashMap, HashSet, and Hashtable.

1. What’s the difference beween int and Integer?

A int is a data type that stores 32 bit signed two's compliment integer. On other hand Integer is a wrapper class which wraps a primitive type int into an object. int helps in storing integer value into memory.

1. What is serialization?

Seialization is the process of turning an object in memory into a stream of bytes so you can do stuff like store it on disk or send it over the network.

Deserialization is the reverse process: turning a stream of bytes into an object in memory.

1. Create List and Map. List A contains 1,2,3,4,10(integer) . Map B contains ("a","1") ("b","2") ("c","10") (key = string, value = string)

Question: get a list which contains all the elements in list A, but not in map B.

1. Implement a group of classes that have common behavior/state as Shape. Create Circle, Rectangle and Square for now as later on we may need more shapes. They should have the ability to calculate the area. They should be able to compare using area. Please write a program to demonstrate the classes and comparison. You can use either abstract or interface. Comparator or Comparable interface.