**Assignment 3**

1. Explain polymorphism.

Polymorphism in Java is the ability of an object to take many forms. To put it simply, polymorphism in Java allows us to perform the same action in many different ways. Any Java object that can pass more than one IS-A test is polymorphic in Java. Therefore, all the Java objects are polymorphic as it has passed the IS-A test for their own type and for the class Object.

1. What is overloading?

Method Overloading is a feature that allows a class to have more than one method having the same name, if their argument lists are different. It is similar to constructor overloading in Java, that allows a class to have more than one constructor having different argument lists.

1. What is overriding?

If subclass (child class) has the same method as declared in the parent class, it is known as method overriding in Java. In other words, If a subclass provides the specific implementation of the method that has been declared by one of its parent class, it is known as method overriding.

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

The parameter aCar could not be reassigned in doSomething(). 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);?

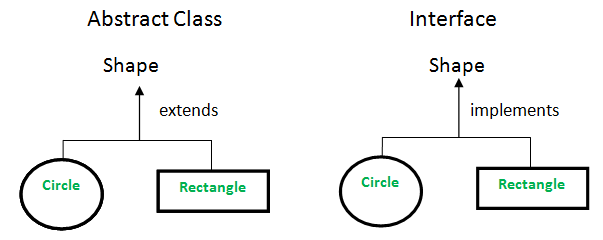
Yes

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

No. Class variables also known as static variables are declared with the static keyword in a class, but outside a method, constructor or a block.

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

As we know that abstraction refers to hiding the internal implementation of the feature and only showing the functionality to the users. i.e. what it works (showing), how it works (hiding). Both abstract class and interface are used for abstraction, henceforth Interface and Abstract Class are required prerequisites



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

Yes, we can declare an abstract class with no abstract methods in Java. An abstract class means that hiding the implementation and showing the function definition to the user. An abstract class having both abstract methods and non-abstract methods.

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

One of the most common uses for constructors in abstract classes is to avoid redundancy.

1. What is a native method?

A native method in Java is a method whose implementation is written in other languages such as c and c++.

The ‘native’ keyword is used before a method to indicate that it is implemented in other language.

1. What is marker interface?

It is an empty interface (no field or methods). Examples of marker interface are Serializable, Cloneable and Remote interface. All these interfaces are empty interfaces.

public interface Serializable

{

// nothing here

}

1. Why to override equals and hashCode methods?

HashMap and HashSet use the hashcode value of an object to find out how the object would be stored in the collection, and subsequently hashcode is used to help locate the object in the collection. Hashing retrieval involves:

* First, find out the right bucket using hashCode().
* Secondly, search the bucket for the right element using equals()

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 between int and Integer?

int is a data type, Integer is Object.

1. What is serialization?

Java provides a mechanism, called object serialization where an object can be represented as a sequence of bytes that includes the object's data as well as information about the object's type and the types of data stored in the object.

After a serialized object has been written into a file, it can be read from the file and deserialized that is, the type information and bytes that represent the object and its data can be used to recreate the 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.

List C contains 3,4(integer)

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.

文本

描述已自动生成

文本

描述已自动生成