

**Resume of thinking in Java**

Name: Sebastian Landazuri

Date: 18/11/2018

The hidden implementation

The objective of the client programmer is to reuse a complete toolbox to be used for the development of applications

The objective of the class creator is to expose the client only what is necessary and keep everything else hidden

In any relationship it is important to maintain limits +

Java access modifiers: public, private, protected

Implementation reuse

The simplest way to reitulize a class is to directly use an object of that class = create a member object

Defining a new class from an existing class is called composition

Heritage

Using inheritance you can build a hierarchy of types that expresses the problem you are trying to solve in terms of their types

all messages that can be sent to the objects of the base class can also be sent to the objects of the derived class

you do not necessarily have to add new methods to the interface

to substitute a method it is enough to create a new definition for it in the derived class

Relationships is a and it is a like

the relationship between the base class and the derived classes is said to be "a"

add new interface elements to a derived type, expanding the interface "it's like a"

interchangeable objects with polymorphism

if it is not known which code fragment will be executed then a new subtype will be added and the code that is executed may be different without it being necessary to perform

changes in the method that calls it

The single root hierarchy

all classes of the last instance should be inherited from a single base class = yes

the flexibility of c ++, only if you have a lot of code in C

it is possible that all the objects of a single root hierarchy have a certain functionality

containers

you do not need to know how much to store a container, just create a container object

two basic types of arraylist and linkedl containers.