**TALLER 3**

**RELACIONES ENTRE OBJETOS**

**Integrantes**

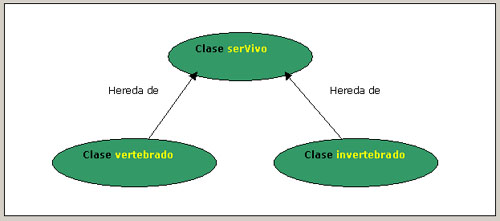
* Natalia Grijalba [ngrijalbah@unal.edu.co](mailto:ngrijalbah@unal.edu.co)
* David Fajardo [dsfajardob@unal.edu.co](mailto:dsfajardob@unal.edu.co)
* Cristian Bernal [crabernalmo@unal.edu.co](mailto:crabernalmo@unal.edu.co)
* Daniel Osuna [dgosunar@unal.edu.co](mailto:dgosunar@unal.edu.co)

1. Responda las siguientes preguntas:
2. **¿Cuál cree que es rol de herencia en un programa de Java?**
3. **¿Cómo la herencia promueve la reutilización de software?**

La herencia es una forma de reutilización de software en la cual, las nuevas clases se crean a partir de clases existentes al absorber sus atributos y comportamientos, y se mejoran con nuevas capacidades, o con modificaciones en las capacidades ya existentes. La herencia promueve la reutilización de software comprobado, depurado y de alta calidad, con lo que reduce los problemas una vez que el sistema se hace funcional.

1. **¿Cómo se podría explicar la Jerarquía (Hierarchy) en la Programación Orientada a Objetos?**

La Jerarquía la podríamos explicar de la siguiente manera: es la propiedad que nos permite la ordenación de las abstracciones.



1. **Explique la diferencia entre composición y herencia. De un ejemplo.**

La diferencia entre composición y herencia es la siguiente:

**HERENCIA:** permite definir una clase a partir de otra.

* Reutilización de caja blanca: los aspectos internos de la superclase son visibles a las subclases.
* Soportada por el lenguaje de programación.
* Estática: se define en tiempo de compilación.

**COMPOSICION:** nueva funcionalidad mediante composición de objetos.

* Reutilización de caja negra: no hay visibilidad de los aspectos internos de los objetos.
* Requiere interfaces bien definidas.
* Dinámica: se define en tiempo de ejecución.

**EJEMPLO:**

1. **En Java una subclase puede heredar de máximo una superclase. En otros lenguajes como c++ es posible que una clase herede de más de una clase (Herencia Múltiple). Explique los pros y contras de esta práctica.**
2. Herencias:
3. **Zoo**
4. **Figuras**

Class Rectangle

Class Rectangle Triangle

Class Parallelogram

Class Isoceles Triangle

Length

Width

Arear

Perimeterr

Base

Heigth

Areart

Perimeterrt

Base

Heigth

Areait

Perimeterit

Length

Width

Areap

Perimeterp

Test Shapes

* ¿Cuáles características tienen en común?

Cada clase necesita de ingresársele dos valores, esta toma los valores y la figura que se está evaluando y calcula el área y perímetro de esta.

* ¿Cómo se crean las clases más generales?

Cada clase se crea con un nombre específico, se creó una clase para cada figura y en esta se definieron las variables seleccionadas para analizar la figura.