

Análisis y Diseño orientado a objetos

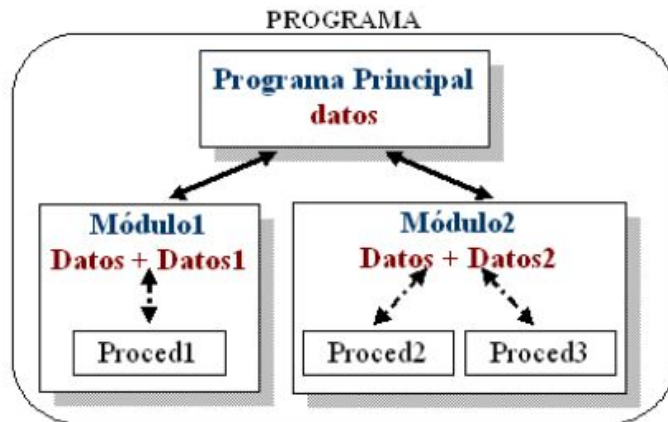


Java | Fundamentos de POO



Modularidad

Es la propiedad que permite tener independencia entre las diferentes partes de un sistema. La modularidad consiste en dividir un programa en módulos o partes, que pueden ser compilados separadamente, pero que tienen conexiones con otros módulos.





Java | Beneficios de la Modularidad

- Programas más simples, ya que puede ser comprendido, verificado, programado, depurado, mejorado y alterado por partes.
- Módulos que pueden ser desarrollados con relativa independencia.
- Disminución de la posibilidad de errores al reducir la complejidad.
- Programas que pueden evaluarse por partes, por lo cual todo el test se hace más fácil.
- Módulos de función única que pueden ser **REUTILIZADOS**.



Java | Herencia



La herencia, es una forma de reutilización del software en la que se crea una nueva clase absorbiendo los miembros de una clase existente. Además, se mejoran con nuevas capacidades, o con modificaciones en las capacidades ya existentes.

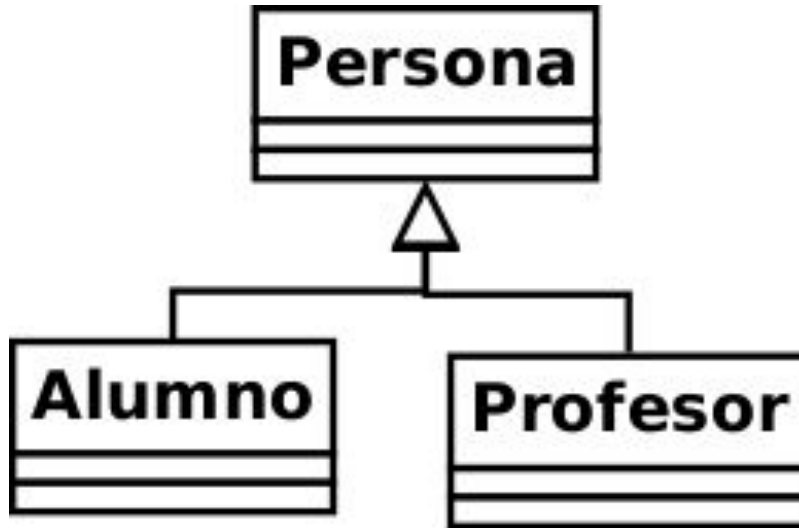
Es el mecanismo por el cual una clase permite heredar las características (atributos y métodos) de otra clase. La herencia permite que se puedan definir nuevas clases basadas de unas ya existentes a fin de reutilizar el código, generando así una jerarquía de clases dentro de una aplicación.



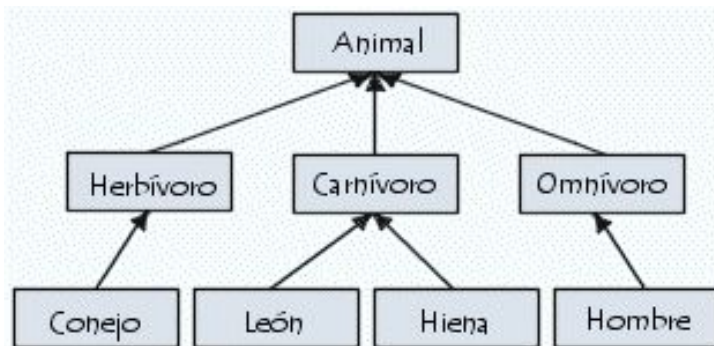
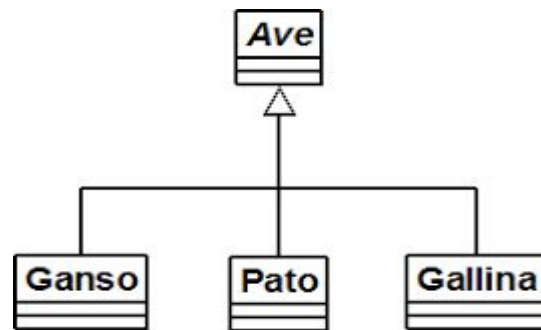
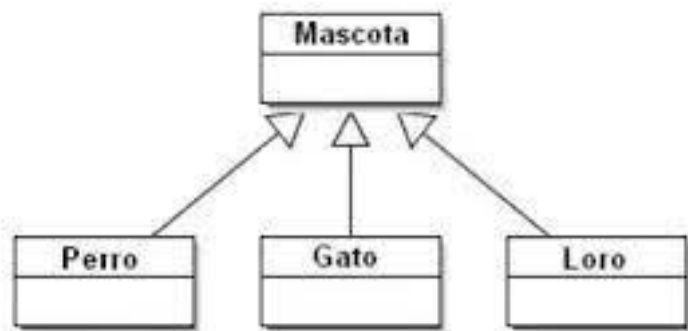
Java | Herencia



En el siguiente ejemplo tenemos 3 clases y se puede ver claramente que, las clases Alumno y Profesor heredan atributos y métodos de la clase persona.



Java | Herencia



Java | Características de la Herencia



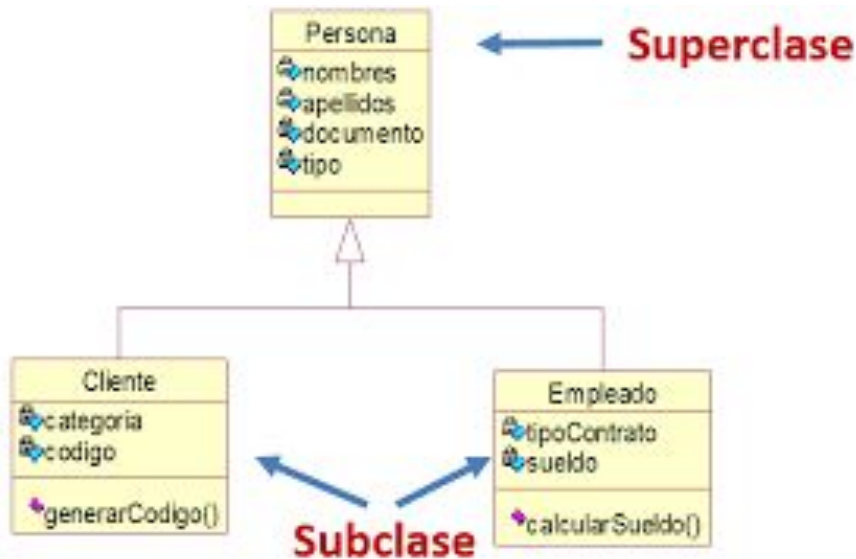
- Es un mecanismo que sirve para reutilizar clases.
- Se utiliza cuando existen clases que comparten muchas de sus características.
- Se extiende la funcionalidad de clases más genéricas.
- Se introducen los conceptos de superclase y subclase.



Java | Herencia Simple



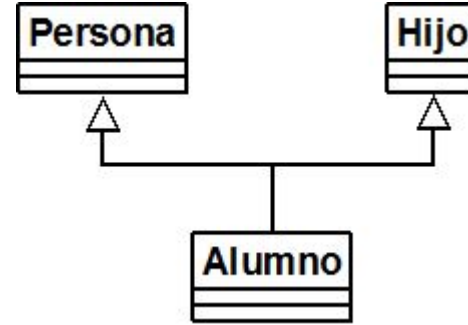
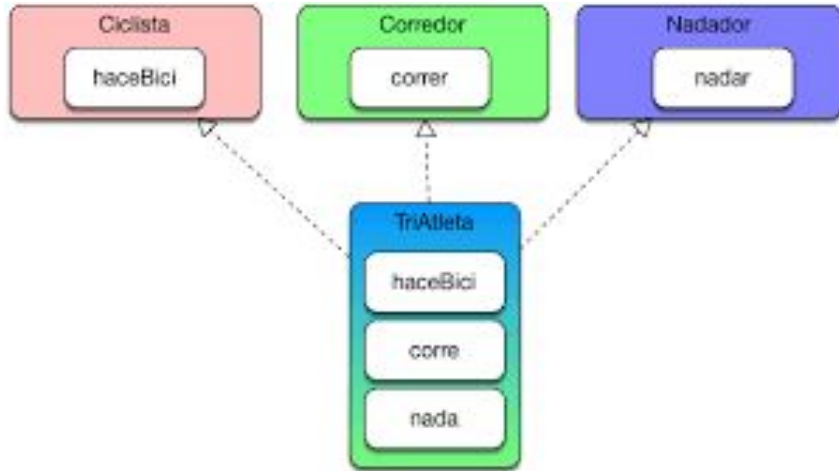
Una clase solo tiene una superclase. Un objeto pertenece a una sola clase.



Java | Herencia Múltiple



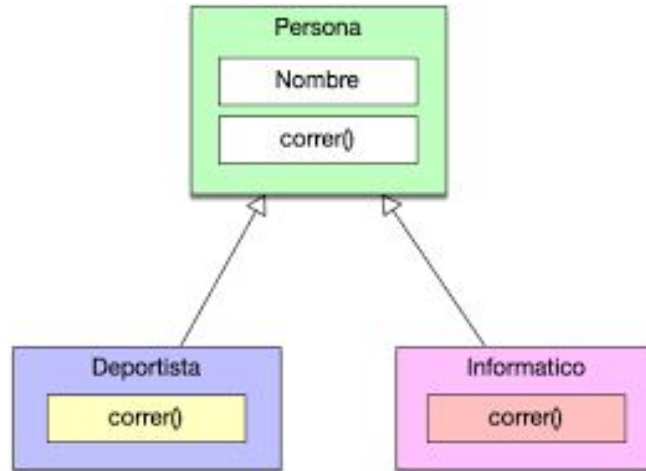
Una clase tiene 2 o más superclases. Un objeto pertenece a más de una clase.



Java | Polimorfismo



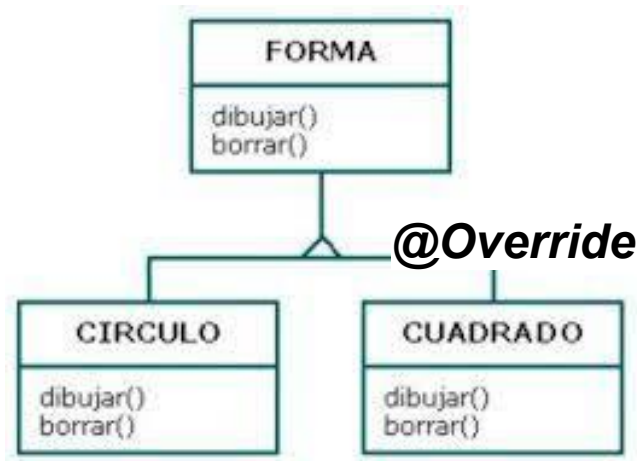
El polimorfismo es un concepto de la Programación Orientada a Objetos (POO) que permite que un objeto pueda tomar múltiples formas.



Java | Implementación de Polimorfismo



En Java, el polimorfismo se logra a través de la herencia y la implementación de interfaces. Se debe utilizar la palabra clave ***extends*** para heredar de una clase base e ***implements*** para implementar una interfaz. Además, se debe utilizar la anotación ***@Override*** al sobrescribir un método de la clase base.





UNJu | Universidad
Nacional de Jujuy



Argentina
programa
4.0

Diseño de Software



Java | Diseño de Software



El diseño de software es una actividad previa a la programación que consiste en pensar cómo dividir el sistema o proceso en módulos o en grupos de líneas de código más pequeños, esto permite tener un modelo de los elementos de software que necesito antes de empezar a programar. Por último, poder planificar las tareas pudiendo asignar a distintos programadores cada módulo o fragmento de código.

- Módulo
- Interface
- Implementación





UNJu | Universidad
Nacional de Jujuy



Argentina
programa
4.0

ADOO - Análisis y Diseño Orientado a Objetos



Java | ADOO



Es un método de análisis que examina los requisitos desde la perspectiva de las clases objetos que se encuentran en el vocabulario del dominio del problema.

El uso de Análisis orientado a objetos puede facilitar mucho la creación de prototipos, y las técnicas de desarrollo evolutivo de software.

Prototipo del Sistema --->Producto Final





Java | Análisis, diseño e implementación

1. El análisis es la fase cuyo objetivo es estudiar y comprender el dominio del problema, es decir, el análisis se centra en responder al interrogante ¿QUÉ HACER?
2. El diseño, por su parte, dirige sus esfuerzos en desarrollar la solución a los requisitos planteados en el análisis, esto es, el diseño se haya centrado en el espacio de la solución, intentando dar respuesta a la cuestión ¿CÓMO HACERLO?
3. Por último, la fase de implementación sería la encargada de la traducción del diseño de la aplicación al lenguaje de programación elegido, adaptando por tanto la solución a un entorno concreto.





Java | Conceptos importantes de la ADOO

En el caso de la POO, existen diferentes metodologías que consisten en construir un modelo (Representación de la realidad a través de diferentes variables) de un dominio de aplicación como:

- OMT: Técnica de Modelado de Objetos.
- UML: Lenguaje Unificado de Modelado.



Modelado de Objetos - UML



Java | UML (Lenguaje Unificado de Modelado)

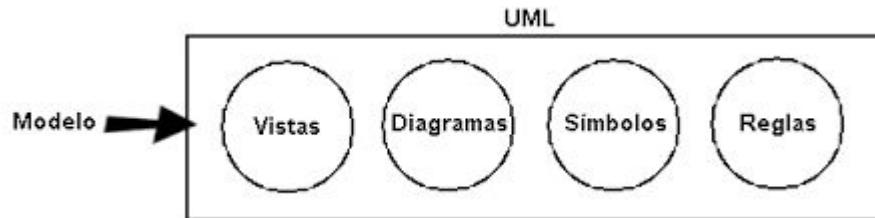


UML modela un sistema mediante el uso de objetos que forman parte de él así como, las relaciones estáticas o dinámicas que existen entre ellos.

Cada símbolo tiene una semántica bien definida.

Cualquier desarrollador puede interpretar un modelo UML sin ambigüedad.

UML facilita la comunicación



Java | Casos de Uso

