



Capítulo 3: Conceptos básicos de Programación Orientada a Objetos

Profesores: Johan Baldeón – David Allasi

INF237 – Lenguaje de Programación Orientada a Objetos

Agenda

- Conceptos básicos de Programación Orientada a Objetos
 - Clase
 - Objeto
 - Encapsulamiento
 - Herencia
 - Polimorfismo

Introducción

- La programación orientada a objetos está basada en Objetos, se modela y se piensa en los Objetos y lo que estos pueden hacer, o lo que se puede hacer con estos.
- Se pueden construir fácilmente nuevas aplicaciones sin mucho esfuerzo.
- La Herencia puede ser una herramienta muy útil en OOP.

Lenguajes POO

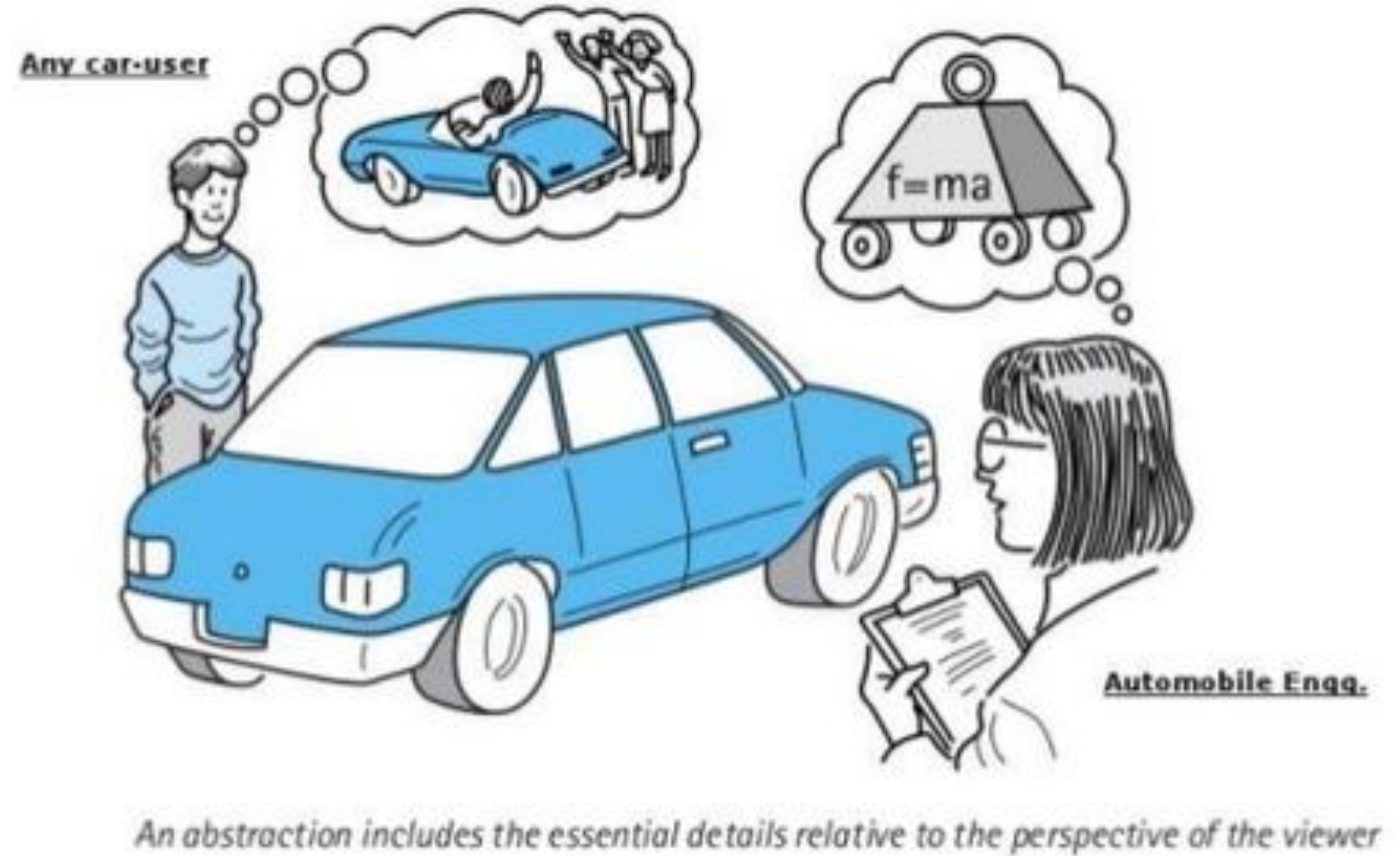
- ADA
- Modula
- Smalltalk (Alan Kay)
- Java
- Eiffel
- C++
- C#
- Object Pascal

Conceptos POO

- Abstracción
- Encapsulamiento
- Ocultamiento de datos
- Clases y Objetos
- Atributos y Métodos
- Herencia
- Polimorfismo

Abstracción

- Se define como la capacidad para examinar algo sin preocuparse de sus detalles internos.
- Existen dos tipos de abstracciones: abstracción de datos y abstracción funcional.



By:-Gourav Kottawar 3/10/2016

Abstracción de Datos

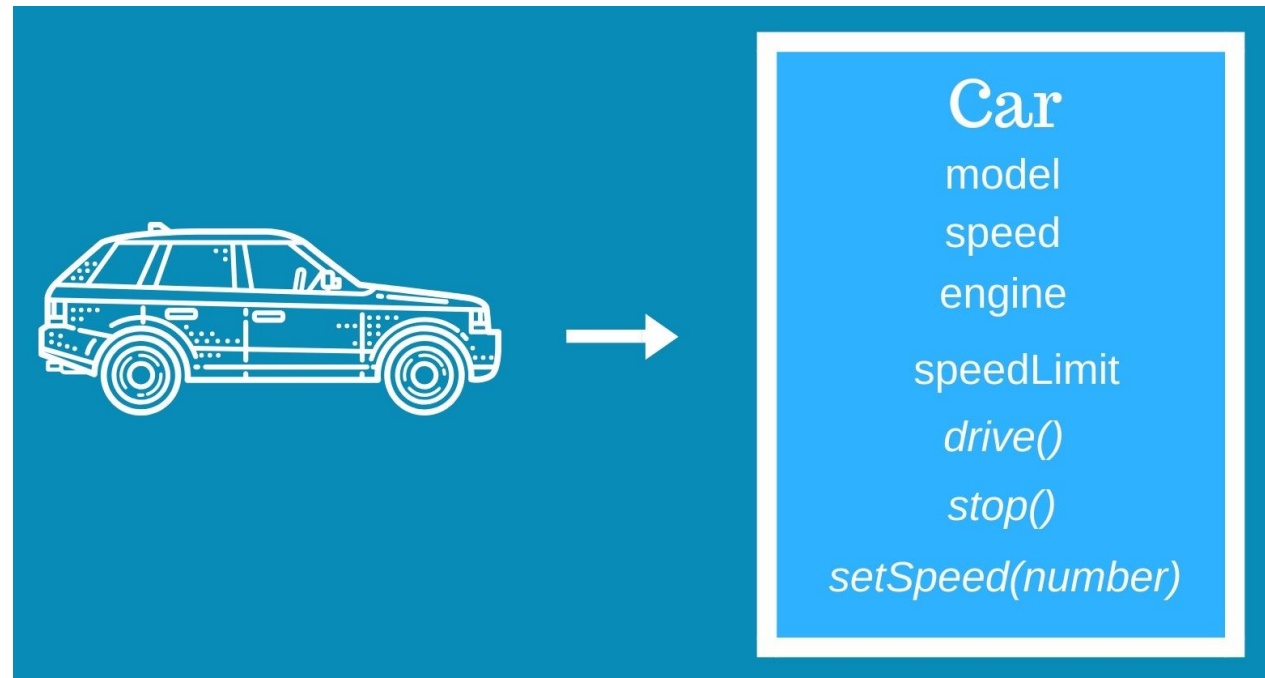
- Consiste en utilizar los datos sin preocuparse por los detalles de su implementación, es decir, lo importante es saber que tipo de información se puede utilizar y no como está almacenado.
- Qué -> Atributo
- Cómo -> Campo

Abstracción Funcional

- Consiste en saber qué es lo que hace un determinado proceso, pero no cómo lo hace.
- Por ejemplo, el proceso Ordenar de la clase Vector, sabemos que el método ordenará los elementos del Vector, pero no nos interesa cómo lo hará.
- Qué -> Método
- Cómo -> Algoritmo

Encapsulamiento

- Consiste en reunir varias cosas para ser manipuladas como una sola unidad.
- Al definir una clase estamos encapsulando los atributos y los métodos.



Abstracción y Encapsulamiento

- La abstracción y el encapsulamiento son conceptos complementarios.
- La abstracción se enfoca a la vista exterior de un objeto.
- El encapsulamiento previene a otros objetos ver su interior, donde el comportamiento de la abstracción se ha realizado.

Ocultamiento de datos

- Consiste en no permitir el acceso a los componentes de una clases (atributos, métodos).
- Esto se logra con la visibilidad que ofrecen los lenguajes de programación.
- Privado: Solo los miembros de la clase tienen acceso.
- Protegido: Solo los miembros de la clase y sus derivados tienen acceso.
- Público: Todos tienen acceso.

Modularidad

- Este concepto viene de la programación modular y se refiere al hecho de realizar un programa por partes, a las cuales se denomina módulos.
- Un módulo es un archivo que contiene un conjunto de declaraciones y/o procesos. En OOP normalmente un módulo contiene la interfaz y la implementación de una o más clases relacionadas.

Clase

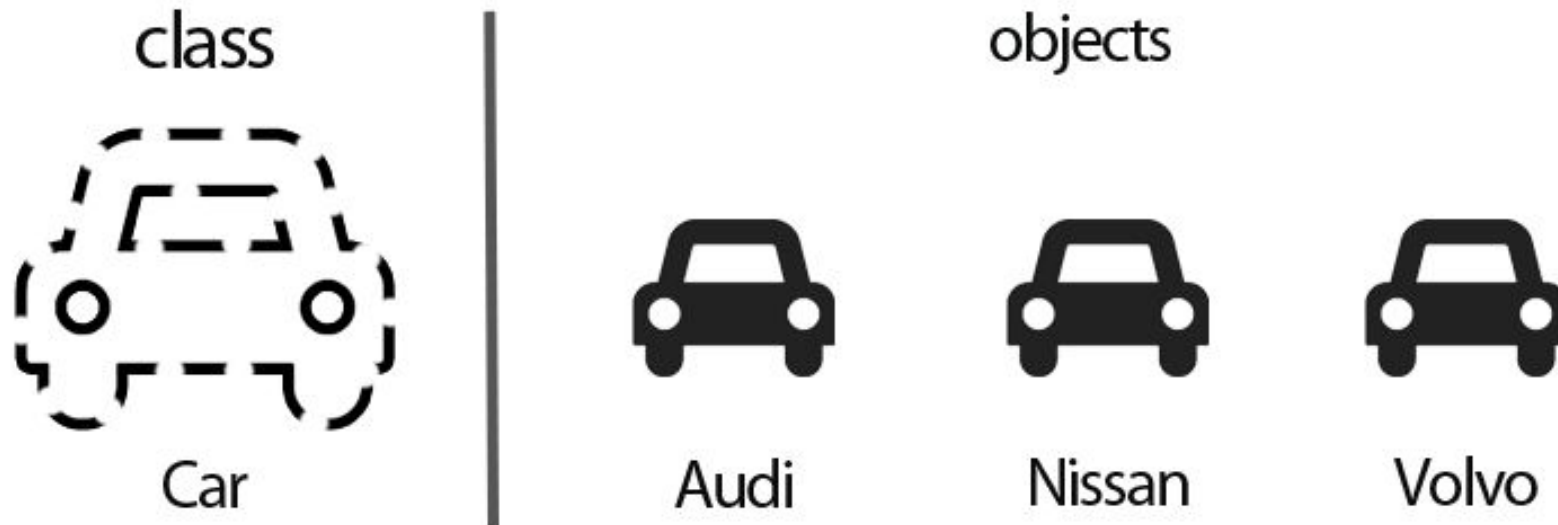
- Una clase representa a un conjunto de objetos que comparten una estructura y un comportamiento comunes.
- La estructura de un objeto se define en su clase, por lo que un objeto es en realidad la instancia de una clase.

Luis Joyanes Aguilar:

- “Una clase es un tipo definido por el usuario (programador) que determina las estructuras de datos y las operaciones asociadas con este tipo”.
- “La clase es simplemente un modelo que se utiliza para describir uno o más objetos del mismo tipo”.

Objeto

- Un objeto es una abstracción de una entidad del mundo real.
- Toda interacción con el mundo externo es por medio de una interface.



Ejemplos de Objetos

- Persona
- Equipo Hardware
- Materiales
- Información
- Software
- Procesos
- Procedimientos

Ejemplos de Objetos

- Objeto Físico: Horno

- Operaciones asociadas:

- Encender
- Apagar
- Cargar
- Descargar

- Objeto Abstracto: Cola

- Operaciones asociadas:

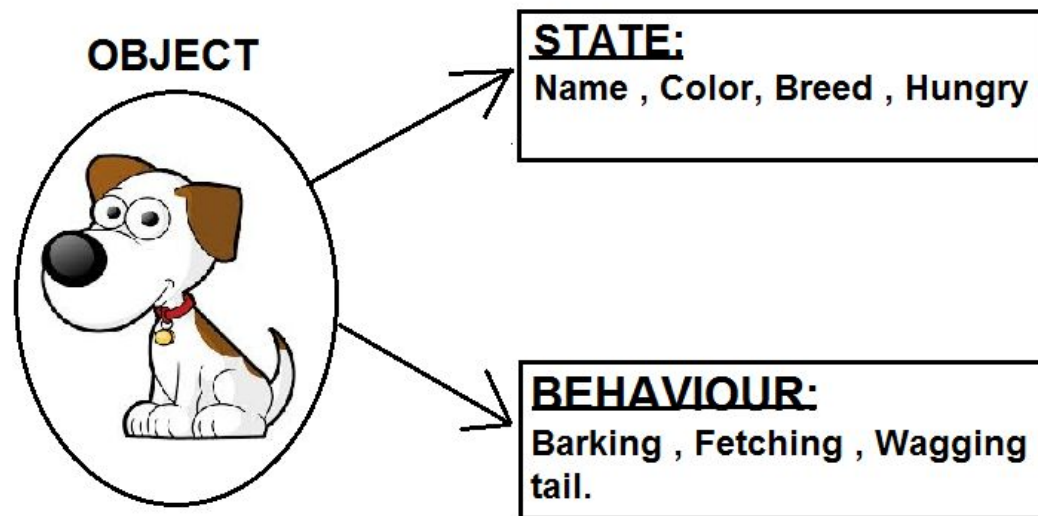
- Agregar
- Eliminar
- Verificar si es vacía
- Primero en la cola

Objeto

- Cada objeto tiene su propia identidad (única) que lo distingue de los demás objetos. En otras palabras, dos objetos distintos no son iguales aunque todos los valores de sus atributos sean idénticos.
- Un objeto tiene un estado y un comportamiento bien definido.

Estado

- El estado de un objeto se compone de todas las propiedades de un objeto (estáticas) y los valores asociados a ellas (dinámicos).
- Estos valores pueden ser: simples o denotar a otro objeto



Atributos

- Características que definen a la clase, es lo que el usuario (de la clase) puede ver. Es la abstracción.
- Variables que están dentro de los objetos y que contienen los valores del objeto.
- También son llamados: propiedades, datos miembro.

Ejemplos de Atributos

- Una fracción está definida por un par de números naturales, los cuales se denominan numerador y denominador.

Fraccion
Numerador
Denominador

- Una fecha está definida por tres datos: día, mes y año.

Fecha
Día
Mes
Año

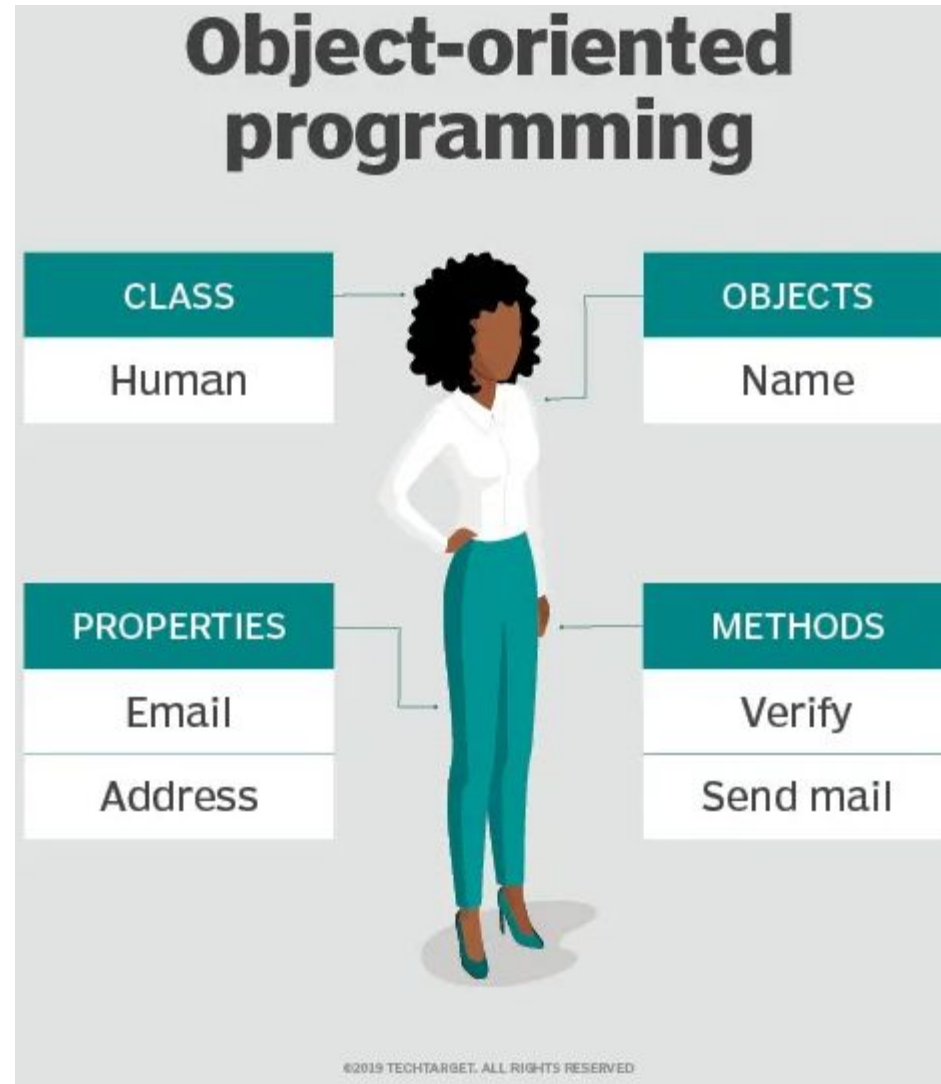
Comportamiento

- El comportamiento de un objeto es la forma en que actúa y reacciona, en términos de sus cambios de estado e intercambio de mensajes con otros objetos.
- El comportamiento de un objeto está determinado por las operaciones de que pueden ser invocadas sobre el mismo.

Métodos

- Son las operaciones que se pueden realizar con los objetos de la clase.
- Son los procesos (funciones o procedimientos) que permiten trabajar con los datos de los objetos.
- También son llamados: funciones miembro.

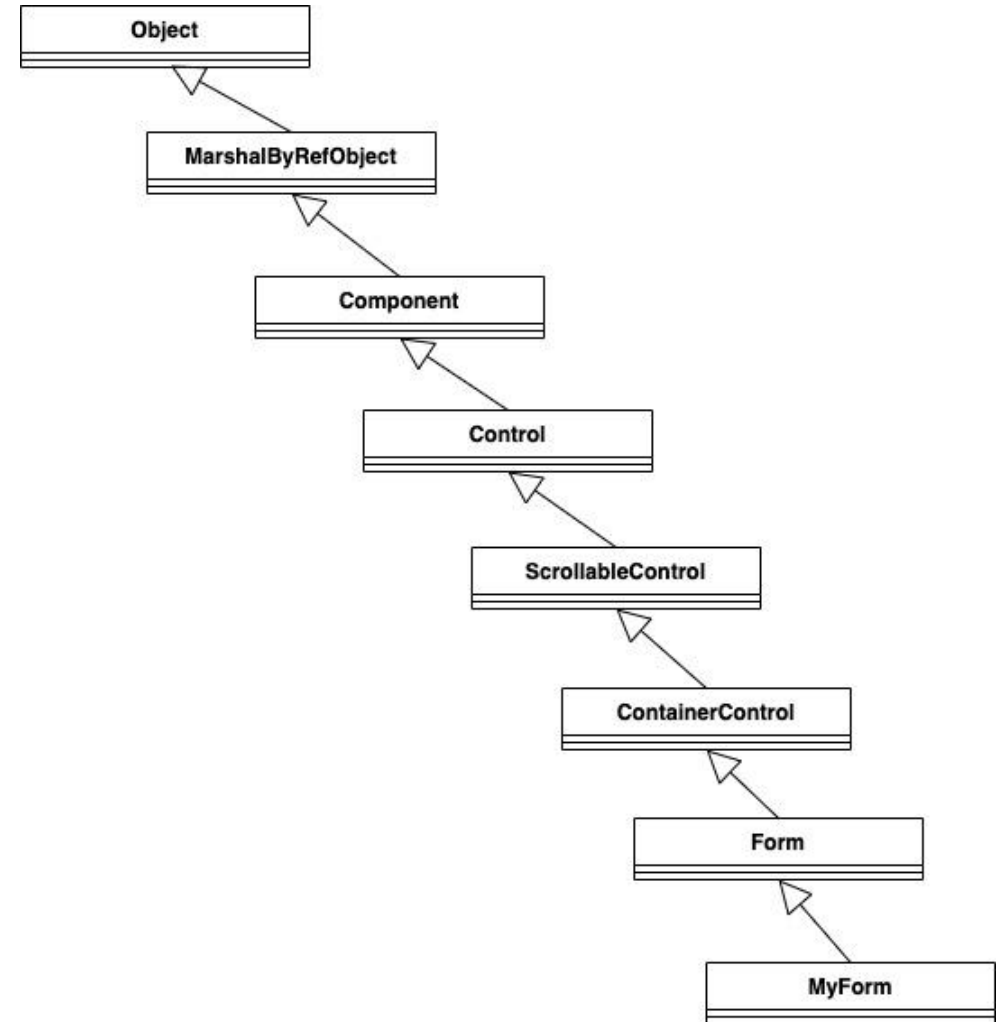
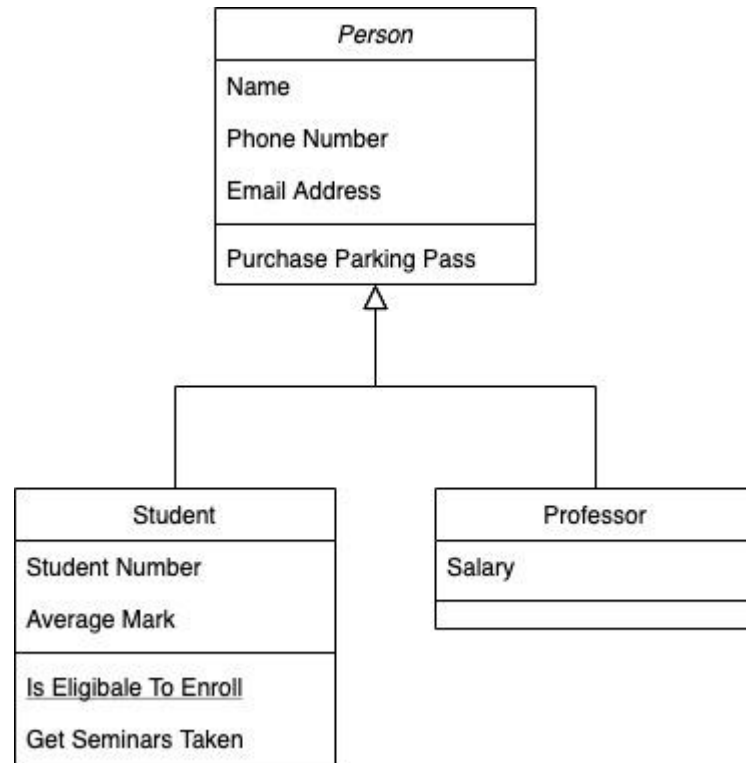
Atributos y Métodos



Herencia

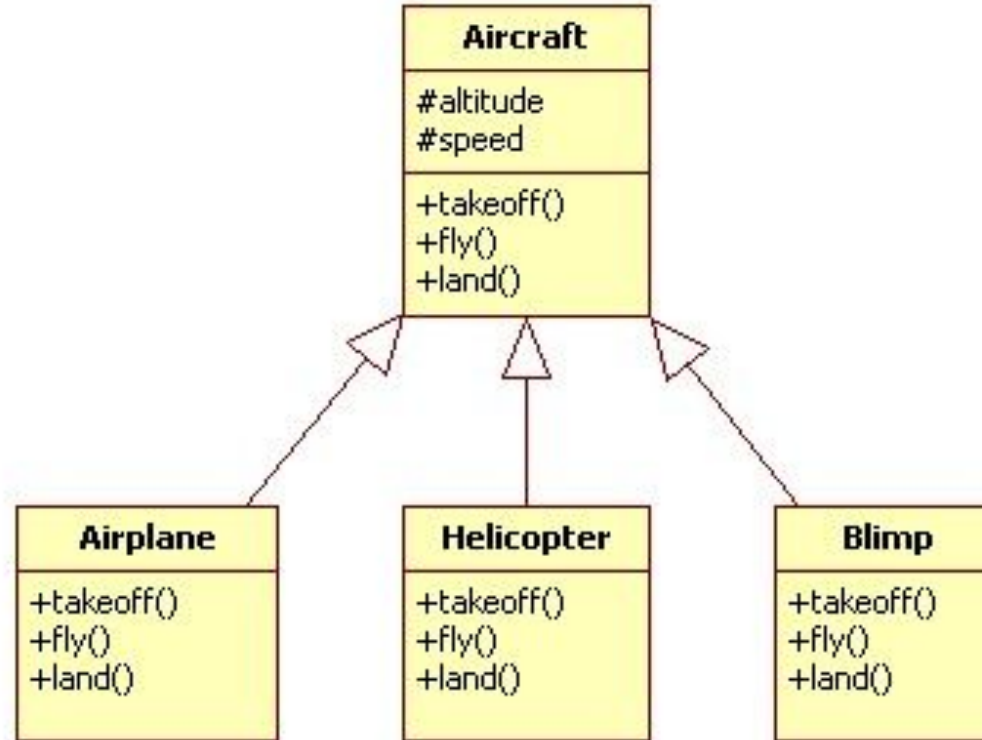
- Consiste en el compartir atributos y métodos entre clases basándose en una relación jerárquica.
- Una clase puede definirse ampliamente y redefinirse sucesivamente en subclases más refinadas.
- Cada subclase que se incorpora, hereda todas las propiedades de su superclase y adiciona sus propias y únicas propiedades.

Ejemplos de Herencia



Polimorfismo

- Significa que la misma operación puede comportarse diferentemente sobre distintas clases. Por ejemplo, la operación “despegar” puede comportarse diferentemente sobre una clase llamada Avión, diferente para una clase Helicóptero y diferente para una clase Dirigible.



Bibliografía

- Heege, M. (2007). *Expert C++/CLI : .NET for Visual C++ programmers*. Berkeley, CA New York: Apress Distributed to the Book trade by Springer-Verlag.
- Horton, I. (2014). *Ivor Horton's beginning Visual C++ 2013*. John Wiley & Sons, Wrox.
- Stroustrup, B. (2014). *A tour of C++*. Pearson Education.
- Stroustrup, B. (2014). *Programming: Principles and Practice Using C++* (2nd ed.). Addison-Wesley.
- Stroustrup, B. (2018). *The C++ programming language* (4th ed.). Addison-Wesley.