
Introducción a la POO.

Contenidos:

- Introducción.
- Abstracción
- Clases.
- Objetos.

Introducción

¿Qué es la programación orientada a objetos? ¿Y un lenguaje orientado a objetos?

- ✓ La programación orientada a objetos es una “filosofía”, un modelo de programación, con su teoría y su metodología.
- ✓ Un lenguaje orientado a objetos es un lenguaje de programación que permite el diseño de aplicaciones orientadas a objetos.

¿Qué es la abstracción?

Supresión intencionada (u ocultación) de algunos detalles de un proceso o artefacto, con el fin de destacar más claramente otros aspectos, detalles o estructuras.

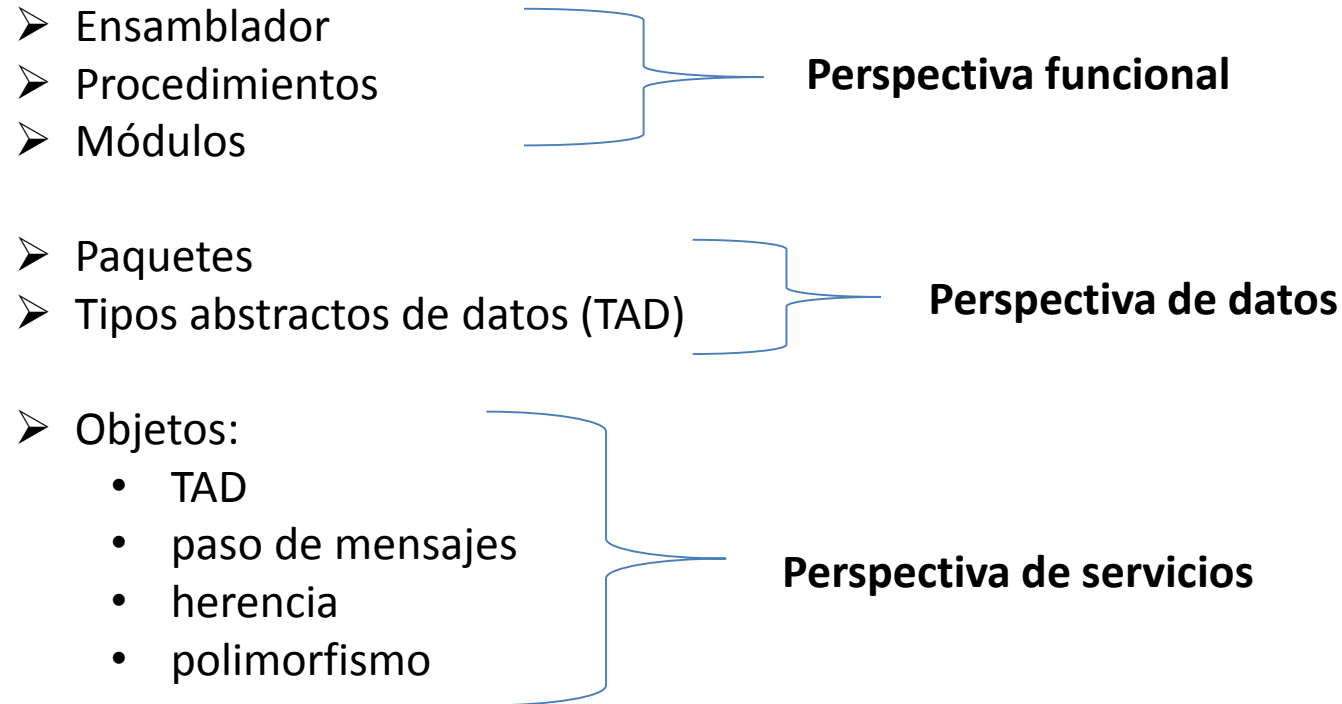
En cada nivel de detalle cierta información se muestra y cierta información se omite => Mapas de Google

Mediante la abstracción creamos **MODELOS** de la realidad.



Niveles de abstracción.

El nivel de abstracción depende del lenguaje de programación utilizado, ya que cada uno proporciona unos mecanismos propios.



Encapsulación

Consiste en la omisión intencionada de detalles en un **desarrollo**.

Cuando esta omisión es interna (objeto) y externa => **ENCAPSULACIÓN**.

Vista externa

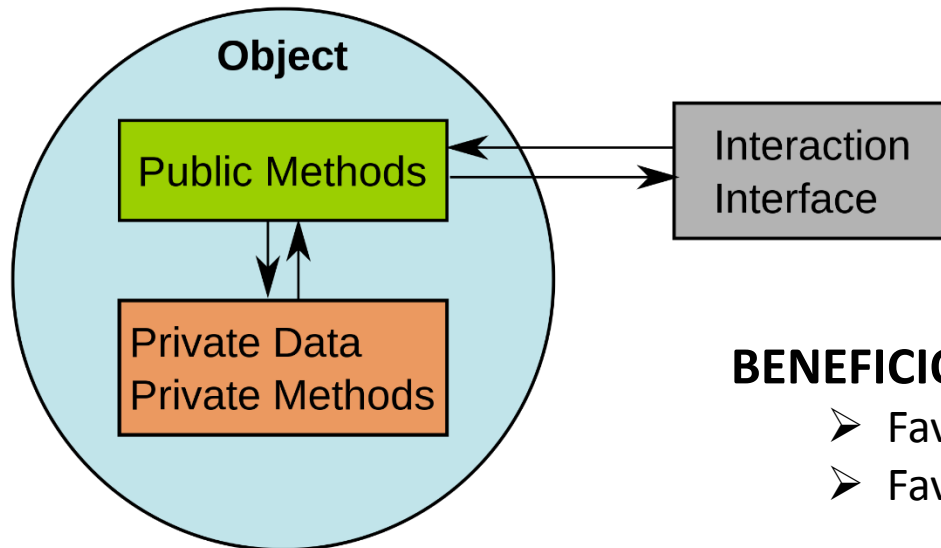
¿QUÉ sabe hacer el objeto?

INTERFAZ

Vista interna

¿CÓMO lo hace?

IMPLEMENTACIÓN



BENEFICIOS:

- Favorece la intercambiabilidad.
- Favorece la comunicación entre miembros del equipo de desarrollo.

Programación Orientada a Objetos

Es una metodología de desarrollo de aplicaciones en la cual éstas se organizan como colecciones cooperativas de **objetos**, cada uno de los cuales representan una instancia de alguna **clase**, y cuyas clases son miembros de **jerarquías de clases** unidas mediante relaciones de **herencia**. (Grady Booch).



Clases

Son una descripción de un conjunto de objetos del mundo real que comparten una estructura y un comportamiento común.

Las **clases** constan de datos y métodos (rutinas) que resumen las características comunes de un conjunto de objetos.

Un programa informático está compuesto por un *conjunto de clases*, a partir de las cuales se crean objetos que interactúan entre sí.

Podemos definir una clase como una plantilla o prototipo en el que se definen:

- Los **atributos** comunes a todos los objetos de la clase.
- Los **métodos** definen el comportamiento de la clase, es decir, las acciones que pueden realizar.

Objetos

Una **Clase**, como hemos visto, no es más que una especificación que define las características y el comportamiento de un determinado tipo de objetos. Piensa en ella como si se tratara de una plantilla, molde o esquema a partir del cual podremos construir **objetos** concretos.

Los objetos tienen un **estado** y un **comportamiento**.

Los objetos tienen unas características fundamentales que los distinguen:

- **Identidad.** Es la característica que permite diferenciar un objeto de otro. De esta manera, aunque dos objetos sean exactamente iguales en sus atributos, son distintos entre sí. Puede ser una dirección de memoria, el nombre del objeto o cualquier otro elemento que utilice el lenguaje para distinguirlos. Por ejemplo, dos vehículos que hayan salido de la misma cadena de fabricación y sean iguales aparentemente, son distintos porque tienen un código que los identifica.
- **Estado:** El estado de un objeto viene determinado por parámetros o atributos que lo describen y los valores de éstos. Por ejemplo, si tenemos un objeto Coche, el estado estaría definido por atributos como Marca, Modelo, Color, Cilindrada, etc.
- **Comportamiento:** Son las acciones que se pueden realizar sobre el objeto. Con respecto al ejemplo del objeto Coche, el comportamiento serían acciones como: arrancar(), parar(), acelerar(), frenar(), etc.

Ejercicios:

Piensa en como serían las siguientes clases (atributos y métodos), crea varios objetos y débátelas con tus compañeros:

- 1) Persona.
- 2) Avión.
- 3) Perro.
- 4) Piso
- 5) Mesa.
- 6) Moto.
- 7) Casa.
- 8) Silla.
- 9) Mamífero.
- 10) Vehículo.
- 11) Mueble.
- 12) Edificio