Objetos

Una vez que hemos hablado de las clases el siguiente paso es hablar de los **objetos**. Los objetos son la clave para entender la programación orientada a objetos. Todo a nuestro alrededor puede ser considerado un objeto.

Consideramos que los objetos de nuestro alrededor tienen 2 características comunes: el **estado** y el **comportamiento**. El estado hace referencia al estado actual de una característica del objeto, pensando en un coche su color, velocidad, marcha,... Mientras que el comportamiento hace referencia a las acciones que el objeto puede llevar a cabo, continuando con el coche: acelerar, frenar, cambiar de marcha,... En ocasiones un objeto puede estar compuesto por otros objetos. Los objetos guardan su estado en campos y exponen su comportamiento a través de métodos. Los métodos son quienes operan con el estado de los objetos con lo que son los encargados de proporcionar la comunicación con el objeto. Al hacer que se acceda a los campos/atributos a través métodos garantizados que en los campos únicamente haya valores que nos interese (lo trataremos en el apartado encapsulación).

Al trabajar con objetos obtenemos una serie de beneficios como:

- Modularidad: Al escribir y mantener el código fuente de cada objeto por separado.
- Ocultación de información: Al hacer únicamente se interacciones con los métodos de un objeto conseguimos ocultar la implementación interna
- Reusabilidad: Cuando trabajemos con la herencia veremos que si tenemos objetos funcionando podemos beneficiarnos de su implementación y evitarnos codificar lo ya hecho.
- Facilidad de uso: Si un determinado objeto nos ocasiona problemas será suficiente con eliminarlo de la aplicación y programar uno que lo reemplace.

Desde un punto de vista técnico, **un objeto es una instancia de una clase**, pero ¿qué significa esto?

Cuando creamos una instancia estamos reservando una zona de memoria dedicada para el objeto en cuestión y, en Java, esa zona va a permanecer ahí mientras la variable en cuestión (u otra) la referencie. Tras leer lo anterior podemos pensar en un objeto como que un puntero a la zona de memoria que ocupa pero en POO no suele hablarse siguiendo esa terminología.

A diferencia de otros lenguajes de programación, como C, **en Java no es necesario liberar las zonas de memoria cuando dejan de utilizarse**. En Java existe un mecanismo llamado

Copyright 2026

recolector de basura que se encarga de ir liberando las zonas de memoria que no están siendo referenciadas por ningún objeto.

Con ánimo de aportar otro punto de vista y definiciones vuelvo a recurrir a la wikipedia:

> En el paradigma de programación orientada a objetos (POO, o bien OOP en inglés), un objeto es una unidad dentro de un programa de computadora que **consta de un estado y de un comportamiento**, que a su vez constan respectivamente de datos almacenados y de tareas realizables durante el tiempo de ejecución. Un objeto puede ser creado instanciando una clase, como ocurre en la programación orientada a objetos(...)
>
> **Estos objetos interactúan unos con otros**, en contraposición a la visión tradicional en la cual un programa es una colección de subrutinas (funciones o procedimientos), o simplemente una lista de instrucciones para el computador. Cada objeto es capaz de recibir mensajes, procesar datos y enviar mensajes a otros objetos de manera similar a un servicio.
>
> En el mundo de la programación orientada a objetos (POO), **un objeto es el resultado de la instanciación de una clase**. Una clase es el anteproyecto que ofrece la funcionalidad en ella definida, pero ésta queda implementada sólo al crear una instancia de la clase, en la forma de un objeto. Por ejemplo: dado un plano para construir sillas (una clase de nombre clase_silla), entonces una silla concreta, en la que podemos sentarnos, construida a partir de este plano, sería un objeto de clase_silla. Es posible crear (construir) múltiples objetos (sillas) utilizando la definición de la clase (plano) anterior. Los conceptos de clase y objetos son análogos a los de tipo de datos y variable(...)
>
> https://es.wikipedia.org/wiki/Objeto_(programaci%C3%B3n)

Revision #1
Created 2022-02-01 11:11:16 CET by Equipo CATEDU
Updated 2022-02-01 11:11:16 CET by Equipo CATEDU

Copyright 2026