3.2 Propiedades de tipos y objetos

¿Qué es un tipo de dato?

Es un atributo de los datos que indica al ordenador (y/o al programador) sobre la clase de datos que se va a trabajar. Esto incluye imponer restricciones en los datos, como qué valores pueden tomar y qué operaciones se pueden realizar.

Un tipo de dato es, en esencia, un espacio en memoria con restricciones

La asignación de tipos a los datos tiene **dos objetivos** principales:

* Por un lado, detectar errores en las operaciones
* Por el otro, determinar cómo ejecutar estas operaciones

La mayor parte de los lenguajes de programación permiten al programador definir tipos de datos adicionales, normalmente combinando múltiples elementos de otros tipos y definiendo las operaciones del nuevo tipo de dato

¿Qué es un objeto?

Un objeto es una unidad dentro de un [programa de computadora](https://es.wikipedia.org/wiki/Programa_%28computaci%C3%B3n%29) 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](https://es.wikipedia.org/wiki/Instancia_%28programaci%C3%B3n%29) una [clase](https://es.wikipedia.org/wiki/Clase_%28inform%C3%A1tica%29), como ocurre en la programación orientada a objetos, o mediante escritura directa de código y la replicación otros objetos, como ocurre en la [programación basada en prototipos](https://es.wikipedia.org/wiki/Programaci%C3%B3n_basada_en_prototipos).

La dificultad que presenta este enfoque es la creación de una representación abstracta, en forma de objetos, de entidades que realmente existen (perro, auto, lámpara eléctrica...) o que existen virtualmente (seguridad social, clima...).

Un objeto se caracteriza por varios conceptos:

Identidad

La identidad es la propiedad que permite diferenciar a un objeto y distinguirse de otros. Generalmente esta propiedad es tal, que da nombre al objeto. Tomemos por ejemplo el "verde" como un objeto concreto de una clase color; la propiedad que da identidad única a este objeto es precisamente su "color" verde. Tanto es así que para nosotros no tiene sentido usar otro nombre para el objeto que no sea el valor de la propiedad que lo identifica.

En [programación](https://es.wikipedia.org/wiki/Programaci%C3%B3n) la identidad de los objetos sirve para comparar si dos objetos son [iguales](https://es.wikipedia.org/wiki/Principio_de_identidad) o no. No es raro encontrar que en muchos [lenguajes de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) la identidad de un objeto esté determinada por la dirección de [memoria](https://es.wikipedia.org/wiki/Memoria_%28inform%C3%A1tica%29) de la [computadora](https://es.wikipedia.org/wiki/Computadora) en la que se encuentra el objeto, pero este comportamiento puede ser variado redefiniendo la identidad del objeto a otra propiedad.

Comportamiento

El comportamiento de un objeto está directamente relacionado con su funcionalidad y determina las operaciones que este puede realizar o a las que puede responder ante [mensajes](https://es.wikipedia.org/wiki/Mensaje) enviados por otros objetos. La funcionalidad de un objeto está determinada, primariamente, por su responsabilidad. Una de las ventajas fundamentales de la [POO](https://es.wikipedia.org/wiki/Programaci%C3%B3n_Orientada_a_Objetos) es la [reusabilidad del código](https://es.wikipedia.org/w/index.php?title=Reusabilidad_%28inform%C3%A1tica%29&action=edit&redlink=1); un objeto es más fácil de reutilizarse en tanto su responsabilidad sea mejor definida y más concreta.

Una tarea fundamental a la hora de diseñar una [aplicación informática](https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_inform%C3%A1tica) es definir el comportamiento que tendrán los objetos de las clases involucradas en la [aplicación](https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_inform%C3%A1tica), asociando la funcionalidad requerida por la aplicación a las clases adecuadas.

Estado

El estado de un objeto se refiere al conjunto de atributos y sus valores en un instante de tiempo dado. El comportamiento de un objeto puede modificar el estado de este. Cuando una operación de un objeto modifica su estado se dice que esta tiene "efecto colateral".

Esto tiene especial importancia en aplicaciones que crean varios [hilos de ejecución](https://es.wikipedia.org/wiki/Hilo_de_ejecuci%C3%B3n). Si un objeto es compartido por varios [hilos](https://es.wikipedia.org/wiki/Hilo_de_ejecuci%C3%B3n) y en el transcurso de sus operaciones estas modifican el estado del objeto, es posible que se deriven errores del hecho de que alguno de los hilos asuma que el estado del objeto no cambiará