

3.A. Introducción y conceptos de la POO

✓ Hecho

2. Fundamentos de la Programación Orientada a Objetos.

2.2. Beneficios.

Según lo que hemos visto hasta ahora, **un objeto es cualquier entidad que podemos ver o apreciar**. El concepto fundamental de la Programación Orientada a Objetos son, precisamente, los objetos. Pero ¿qué beneficios aporta la utilización de objetos? Fundamentalmente la posibilidad de representar el problema en términos del mundo real, que como hemos dicho están más cercanos a nuestra forma de pensar, pero existen otra serie de ventajas como las siguientes:

- **Comprensión.** Los conceptos del espacio del problema se hayan reflejados en el código del programa, por lo que la mera lectura del código nos describe la solución del problema en el mundo real.
- **Modularidad.** Facilita la modularidad del código, al estar las definiciones de objetos en módulos o archivos independientes, hace que las aplicaciones estén mejor organizadas y sean más fáciles de entender.
- **Fácil mantenimiento.** Cualquier **modificación en las acciones** queda **automáticamente reflejada en los datos**, ya que ambos están estrechamente **relacionados**. Esto hace que el **mantenimiento de las aplicaciones**, así como su **corrección y modificación** sea mucho más **fácil**. Por ejemplo, podemos querer utilizar un algoritmo más rápido, sin tener que cambiar el programa principal. Por otra parte, al estar las aplicaciones mejor organizadas, es más fácil localizar cualquier elemento que se quiera modificar y/o corregir. Esto es importante ya que se estima que los **mayores costes de software** no están en el proceso de desarrollo en sí, sino en el **mantenimiento** posterior de ese software a lo largo de su vida útil.
- **Seguridad.** La probabilidad de cometer errores se ve **reducida**, ya que no podemos modificar los **datos de un objeto directamente**, sino que debemos hacerlo mediante las **acciones definidas para ese objeto**. Imaginemos un objeto lavadora. Se compone de un motor, tambor, cables, tubos, etc. Para usar una lavadora no se nos ocurre abrirla y empezar a manipular esos elementos, ya que lo más probable es que se estropee. En lugar de eso utilizamos los programas de lavado establecidos. Pues algo parecido con los objetos, no podemos manipularlos internamente, sólo utilizar las acciones que para ellos hay definidas.
- **Reusabilidad.** Los objetos se definen como **entidades reutilizables**, es decir, que los **programas que trabajan** con las mismas estructuras de información, **pueden reutilizar las definiciones de objetos empleadas en otros programas**, e incluso las **acciones definidas** sobre ellos. Por ejemplo, podemos crear la definición de un objeto de tipo **persona** para una aplicación de negocios y deseamos construir a continuación otra aplicación, digamos de educación, en donde utilizamos también personas, no es necesario crear de nuevo el objeto, sino que por medio de la reusabilidad podemos utilizar el tipo de objeto **persona** previamente definido.



Imagen extraída de curso Programación del MECD.

◀ [Orientaciones para el alumno UT03.](#)

Ir a...

[3.B. Objetos y clases.](#) ▶