

¿Qué es la herencia?

En programación orientada a objetos, la herencia es un mecanismo que permite crear una nueva clase basada en una clase existente. La nueva clase hereda atributos y métodos de la clase existente y puede agregar o modificar funcionalidades según sea necesario.

Algunos beneficios de la herencia:

1. **Organización del código:** Facilita la estructuración y organización del código al agrupar características comunes en una clase base.
2. **Mantenimiento simplificado:** Si necesitas realizar cambios, solo tienes que hacerlos en la clase base, y estos cambios se reflejarán automáticamente en todas las clases derivadas.
3. **Facilita la creación de nuevas clases:** Permite crear nuevas clases de manera eficiente al aprovechar y construir sobre clases existentes.

Tipos de herencia:

1. Herencia Simple:

Una clase derivada hereda de una única clase base. Es el tipo más básico de herencia y se utiliza para establecer una relación de "es un/a".

2. Herencia Múltiple:

Una clase derivada puede heredar de más de una clase base. Permite a la clase derivada adquirir características de múltiples fuentes.

3. Herencia Multinivel:

Las clases están organizadas en niveles jerárquicos, donde una clase deriva de otra, y luego otra clase deriva de la segunda. Es útil para modelar relaciones más complejas.

4. Herencia Jerárquica:

Varias clases derivan de una única clase base. Cada clase derivada puede tener características específicas.

5. Herencia Híbrida:

Todas las clases derivadas utilizan un método con la misma firma, pero pueden tener implementaciones diferentes. Es común en situaciones donde se espera un comportamiento similar.

Fuentes de información:

<https://ifgeekthen.nttdata.com/es/herencia-en-programacion-orientada->

Objetos.

ChatGPT.

Aplicaciones utilizadas:

Canva.

Word.

Eclipse.