

Taller de conceptos y principios de programación orientada a objetos

Danny Julián Perilla Mikán

Mayo, 2024

1 Introducción

El paradigma de programación orientada a objetos tuvo su inicio en los años 70, pero su estandarización se logró alrededor de 1997 con la aparición del Lenguaje Unificado de Modelado (UML). Desde los inicios de la programación, siempre se ha pretendido que la tarea de realizar programas de software sea cada vez más sencilla, flexible y portable. En este sentido, la programación orientada a objetos permite crear programas mucho más complejos.

2 Glosario

Los entes más básicos de la programación orientada a objetos (POO) son (1) las clases y (2) los objetos. Estos a su vez se caracterizan por (3) los atributos y (4) los métodos.

- 1 **Clases:** Las clases nos permiten empaquetar atributos y métodos de un individuo o cosa con carácter abstracto, de modo que podemos instanciar la clase en objetos particulares.
- 2 **Objetos:** Es la instancia de una clase, es decir, es una particularización de una idea abstracta representada por la clase.
- 3 **Atributos:** Son los valores que una clase puede almacenar y que determinan las características de los objetos que instancian a la clase.
- 4 **Métodos:** Son las funciones internas de una clase determinada que devuelven algún valor relevante relacionado con la clase.

Los pilares básicos de la programación orientada a objetos son (1) abstracción, (2) encapsulamiento, (3) herencia y (4) polimorfismo, cuyas definiciones damos a continuación:

- 1 **Abstracción:** La abstracción consiste en extraer las características invariantes de un grupo de elementos de una determinada clase o tipo.
- 2 **Encapsulamiento:** El encapsulamiento consiste en dotar a una determinada clase con atributos y métodos característicos que pueden ser públicos, privados o protegidos.
- 3 **Herencia:** La herencia consiste en la dependencia de una subclase respecto a una superclase que contiene atributos y métodos necesarios para la subclase, heredándolos dentro de sus características básicas.
- 4 **Polimorfismo:** El polimorfismo es la capacidad que tiene un objeto para ser instancia de una clase u otra que comparten el mismo nombre, basado solamente en el contexto dado por sus argumentos.

3 Conclusiones

La programación orientada a objetos ha demostrado ser un paradigma robusto y útil para las aplicaciones digitales de la actualidad. Es probable que lo sigamos usando por muchos años más y continúe siendo muy relevante.