



# Programing Paradigms - P00

29-11-2022

—

Leandro Baier

## Metas

Temas a revisar:

1. Conceptos (Paradigma OOP, Objetos, Clases, Abstracción, Herencia, Polimorfismo, Encapsulamiento, etc)
2. Ventajas & Desventajas de OOP
3. Sintaxis Lenguajes de Programación Orientados a Objetos (Java, Python, Golang, C++, C#, Ruby)
4. Programación Orientada a Objetos con 2 a 5 años de experiencia (en al menos uno de los lenguajes del nivel beginner)

## 1. Conceptos (Paradigma OOP, Objetos, Clases, Abstracción, Herencia, Polimorfismo, Encapsulamiento, etc)

La Programación Orientada a objetos permite que el código sea reutilizable, organizado y fácil de mantener. Sigue el principio de desarrollo de software utilizado por muchos programadores DRY (Don't Repeat Yourself), para evitar duplicar el código y crear de esta manera programas eficientes. Además, evita el acceso no deseado a los datos o la exposición de código propietario mediante la encapsulación y la abstracción.

- Objetos: Instancia de una clase
- Clases: Una clase es una plantilla, define de manera genérica como van a ser los objetos de un determinado tipo.
- Abstracción: Es cuando el usuario interactúa solo con los atributos y métodos seleccionados de un objeto, utilizando herramientas simplificadas de alto nivel para acceder a un objeto complejo.
- Herencia: La herencia define relaciones jerárquicas entre clases, de forma que atributos y métodos comunes puedan ser reutilizados. Las clases principales extienden atributos y comportamientos a las clases secundarias.
- Polimorfismo: Es la capacidad de presentar la misma interfaz para diferentes formas subyacentes o tipos de datos. Al utilizar la herencia, los objetos pueden anular los comportamientos principales compartidos, con comportamientos secundarios específicos. El polimorfismo permite que el mismo método ejecute diferentes comportamientos de dos formas: anulación de método y sobrecarga de método.
- Encapsulamiento: La encapsulación contiene toda la información importante de un objeto dentro del mismo y solo expone la información seleccionada al mundo exterior.

## 2. Ventajas & Desventajas de OOP

Ventajas:

- Los componentes se pueden reutilizar.
- Facilidad de mantenimiento y modificación de los objetos existentes.
- Una estructura modular clara se puede obtener, la cual no revelará el mecanismo detrás del diseño.
- Se proporciona un buen marco que facilita la creación de rica interfaz gráfica de usuario aplicaciones (GUI).
- Se acopla bien a la utilización de bases de datos, debido a la correspondencia entre las estructuras.

Desventajas:

- Limitaciones del programador: Es posible que el programador desconozca algunas características del paradigma y de hecho siga utilizando el paradigma estructurado.
- No hay una forma única de resolver los problemas. Esto puede llevar a que diferentes interpretaciones de la solución planteada emerjan.
- Se requiere una documentación amplia para determinar la solución planteada.

## 3, 4. Sintaxis Lenguajes de Programación Orientados a Objetos (Java, Python, Golang, C++, C#, Ruby), Programación Orientada a Objetos con 2 a 5 años de experiencia (en al menos uno de los lenguajes del nivel beginner)

Tengo 2 años de experiencia en GL trabajando con Java.