

# Programación Orientada a Objetos Corte 1

Evelyn León

## 1.1 ¿En la programación orientada a objetos que es el constructor y que función cumple?

Un constructor es un método especial en POO que se utiliza para iniciar una nueva instancia de una clase. Este método se ejecuta automáticamente al crear un objeto y permite establecer el estado inicial de sus atributos o realizar configuraciones necesarias. El nombre del constructor suele coincidir con el nombre de la clase.

## 1.2 ¿En la programación orientada a objetos que es un objeto y que función cumple?

Un objeto cuenta con su estructura interna que combina variables, funciones y estructuras de datos. un objeto es una instancia de una clase. Una clase define la estructura y el comportamiento de un objeto, mientras que el objeto es una instancia concreta de esa clase con un estado y un comportamiento específico.

## 1.3 ¿En la programación orientada a objetos que es una clase y que función cumple?

Son modelos o planos que se utilizan para crear instancias individuales llamadas objetos. Cada objeto creado a partir de una clase puede tener sus propios valores para los atributos de la clase, pero comparte los mismos métodos.

## 1.4 ¿Qué diferencias destacadas se aprecian entre la POO y la programación estructurada (secuencial)?

**Enfoque:** La POO se centra en los objetos y su interacción, mientras que la programación estructurada se enfoca en la secuencia de instrucciones.

**Reutilización de código:** La POO promueve la reutilización a través de la encapsulación y la herencia, mientras que la programación estructurada se basa en la división de un problema en subproblemas más pequeños.

**Organización del código:** La POO proporciona una estructura clara para los programas, lo que facilita la gestión, escalabilidad y mantenimiento del código. La programación estructurada puede volverse compleja y difícil de gestionar para aplicaciones más grandes.

**Flexibilidad:** La POO ofrece mayor flexibilidad y extensibilidad, permitiendo la creación de nuevas clases y la implementación de polimorfismo. La programación estructurada tiene menor flexibilidad y es menos extensible.

**Mantenimiento:** La POO es más fácil de mantener y actualizar, mientras que la programación estructurada puede ser más difícil en proyectos grandes.

## 1.5 ¿Para qué y por qué se usan modificadores de acceso?

Los modificadores de acceso en Java controlan el nivel de visibilidad o accesibilidad de clases, métodos y variables en un programa. Estos modificadores son esenciales para implementar el principio de encapsulamiento.

**Public:** El miembro (clase, método o atributo) es accesible desde cualquier parte del programa. Esto significa que no hay restricciones sobre su acceso, lo que permite que cualquier otra clase pueda interactuar con él.

**Private:** El miembro es accesible solo dentro de la clase en la que está declarado. Esto es útil para proteger los datos y asegurar que no sean modificados directamente desde fuera de la clase, promoviendo así el encapsulamiento.

**Protected:** El miembro es accesible dentro de la misma clase y por las subclases, incluso si estas están en diferentes paquetes. Esto permite que las subclases hereden y accedan a los miembros protegidos de la clase base.

**Default (packageprivate):** Si no se especifica ningún modificador, el miembro es accesible solo dentro del mismo paquete. Esto significa que solo las clases que pertenecen al mismo paquete pueden acceder a esos miembros.