

## **Concepto de Inyección de Dependencias**

La inyección de dependencias es un patrón de diseño que permite a los objetos recibir sus dependencias desde el exterior, en lugar de crearlas por sí mismos. Esto mejora la modularidad del código, facilita las pruebas unitarias y promueve un acoplamiento débil entre los componentes del sistema.

En Java, la inyección de dependencias se puede lograr a través del constructor, métodos setters o incluso la inyección directa de campo. El uso de interfaces para las dependencias permite cambiar fácilmente las implementaciones sin modificar el código dependiente.

## **Explicación del Ejemplo**

En el ejemplo proporcionado, se muestra cómo implementar la inyección de dependencias en Java utilizando interfaces y constructores.

1. Interfaz `MessageService`: Define el método `sendMessage()` que todas las implementaciones deben proporcionar.
2. Implementaciones `EmailService` y `SMSService`: Implementan la interfaz `MessageService`, proporcionando lógica específica para enviar mensajes por correo electrónico y SMS, respectivamente.
3. Clase `NotificationService`: Depende de un objeto `MessageService` para enviar notificaciones. La dependencia se inyecta a través del constructor, lo que permite que `NotificationService` sea independiente de las implementaciones específicas de `MessageService`.
4. Clase `Main`: Muestra cómo se inyectan diferentes implementaciones de `MessageService` (`EmailService` y `SMSService`) en `NotificationService`.
5. Beneficio: La inyección de dependencias permite cambiar las implementaciones de las dependencias sin modificar la clase que las utiliza, lo que facilita la escalabilidad, el mantenimiento y las pruebas del código.