# Optimización y documentación

## Patrones de Refactorización

| **Nombre** | **Descripción** |
| --- | --- |
| **Renombrar** | Cambia el nombre de una clase, interfaz, variable o método a algo más significativo y actualiza todo código fuente del proyecto para reflejar este cambio. |
| **Mover clase** | Mueve una clase a otro paquete o dentro de otra clase. Además, todo el código fuente del proyecto es actualizado para hacer referencia a la clase en el nuevo paquete. |
| **Copiar clase** | Copia una clase en el mismo paquete o en uno diferente. |
| **Eliminar de forma segura** | Elimina, asegurándose de que no van a quedar referencias perdidas al código eliminado. |
| **Ascender** | Mueve métodos y campos a una superclase de la que heredaría la clase actual. |
| **Descender** | Mueve clases, métodos y campos internos a una subclase de la clase actual. |
| **Encapsular campos** | Genera métodos get y set para un campo y, opcionalmente, actualiza todas las referencias a ese campo usando los métodos get y set. |
| **Cambiar firma de un método** | Agregar, eliminar, modificar o cambiar el orden de los parámetros de un método o cambiar el modificador de acceso (público, privado, protegido). Cambiar el nombre del método. |
| **Pasar del nivel interior al exterior** | Mueve una clase miembro hacia un nivel exterior. No confundir con la herencia. Se refiere a clases internas, que pueden heredar de la externa o no. |
| **Hacer estático** | Transforma en estático un método. |
| **Migración de tipo** | Cambia el tipo de un atributo o método. |
| **Extraer interfaz** | Crea una nueva interfaz formada a partir del método público no estático seleccionado en una clase o interfaz. |

Haz los [ejercicios de patrones de refactorización](#_heading=h.30j0zll).

## Generación de código

| **Nombre** | **Descripción** |
| --- | --- |
| **Constructor** | Crea un constructor y permite elegir los parámetros que recibe. |
| **Getters** | Crea los getters de los atributos que se seleccionen. |
| **Setters** | Crea los setters de los atributos que se seleccionen. |
| **Getters y Setters** | Crea los getters y setters de los atributos que se seleccionen. |
| **Sobreescribir métodos**  **Override methods** | Sobreescribe los métodos que se seleccionen. Permite elegir los métodos de todas las clases de las que hereda. |
| **toString()** | Sobreescribe el método to string y permite seleccionar los atributos. |
| **Test** | Permite generar una clase para realizar tests unitarios. |

Haz los [ejercicios de generación de código](#_heading=h.1fob9te).

# Ejercicios

## Patrones de refactorización

### Busca y escribe el nombre en inglés de cada patrón de refactorización.

## Generación de código

### Busca y escribe el nombre en inglés de cada patrón de generación de código

# Bibliografía

<https://manuais.pages.iessanclemente.net/plantillas/DUAL/cd/ud04/1.refactorizar/7.refactorizarjava/index.html>