

# Patrones de Diseño

## Patrones de Creación

### Abstract Factory

Interfaz para crear familias de objetos relacionados o dependientes entre sí, sin especificar clases concretas

### Builder

Separa la construcción de un objeto complejo de su representación. Un mismo objeto puede crear diferentes representaciones

### Factory Method

Define la interfaz para crear un objeto, pero delega en sus subclases la creación de objetos.

### Prototype

Una instancia prototípica especifica los tipos de objetos a crear. Se crean nuevos objetos copiando ese prototipo

### Singleton

Una clase sólo tendrá una instancia y un punto de acceso global a ella

## Patrones Estructurales

### Adapter

Permite que objetos con interfaces a priori incompatibles entre sí puedan trabajar entre ellos

### Bridge

Separa una abstracción de su implementación, pudiendo variar de forma independiente

### Composite

Compone objetos en estructuras de árbol. Permite tratar de manera uniforme los objetos individuales y compuestos

### Decorator

Asigna responsabilidad es dinámicamente. "Herencia flexible"

### Flyweighth

Permite un gran número de objetos de forma eficiente mediante modularidad

## Patrones de Comportamiento

### Command

Encapsula una petición en un un objeto, permitiendo un mayor control de las peticiones

### Interpreter

Permite la representación de la gramática e interpretación de un lenguaje para interpretar sentencias del mismo

### Iterator

Proporciona un acceso secuencial a los elementos de un objeto sin exponer su representación interna

### Observer

Define una dependencia de uno a muchos entre objetos, con actualización automática de cambio de estado

### State

Permite que un objeto cambie de comportamiento cada vez que cambie de estado

### Template Method

Define en una operación el esqueleto del algoritmo, delegando, en subclases, pasos

### Visitor

Representa una operación sobre los elementos de una estructura de objetos. Define nuevas operaciones sin cambiar las clases de los elementos sobre los que opera

### Cadena de Responsabilidad

Evita acoplar el emisor de una petición a su receptor, dando a más de un objeto la posibilidad de respuesta a esa petición