¿Qué es un Patrón?

## ¿Qué es un Patrón?

La regular y repetida manera en la que algo pasa o es hecho.

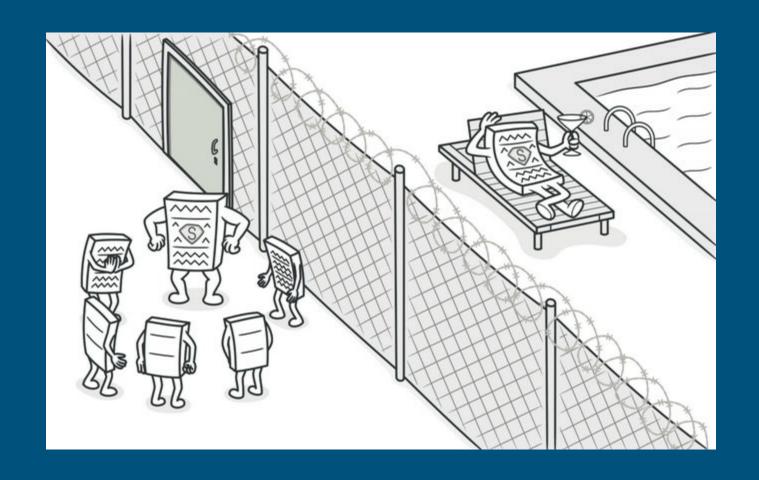
¿Qué es un Patrón de Diseño?

# ¿Qué es un Patrón de Diseño?

Un patrón de diseño es una solución general y reutilizable para problemas típicos y recurrentes en el desarrollo de software. Éstos, dado un contexto similar, nos aportan una solución ya probada y documentada

# Proxy

Patrón de diseño

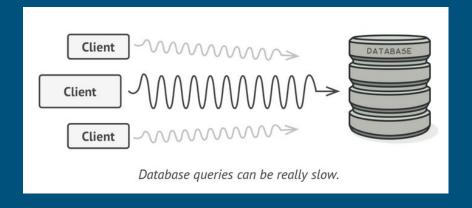


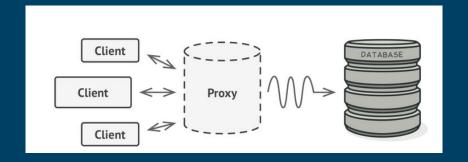
### Patrón Proxy

Es un patrón de diseño estructural que te permite proveer un sustituto o placeholder de un objeto. El proxy controla el acceso al objeto original, permitiendo realizar algo ya sea antes o después de que la petición llegue al objeto original.

## Problema

## Solución



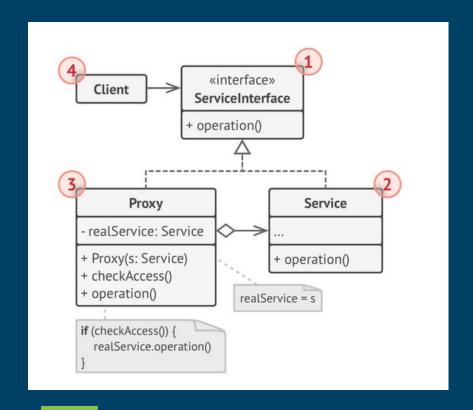


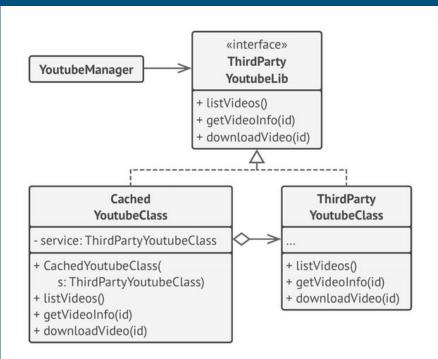
#### Aplicaciones frecuentes/populares

- Proxy remoto: Controla el acceso a un objeto remoto (en un servidor remoto por ejemplo).
- Proxy virtual: Controla el acceso a un recurso cuya creación es costosa. (Inicialización diferida)
- Proxy de protección: Controla el acceso a un recurso en base a unos derechos de acceso.
- Proxy de caché: Cuando se quiere cachear los resultados de peticiones de un cliente
- Proxy de logs: Cuando se quiere mantener una historia de las peticiones al objeto real

#### Estructura

- Interfaz del servicio. El proxy debe tener esta misma interfaz para poder disfrazarse como el objeto Servicio.
- 2. El servicio es una clase que provee alguna lógica de negocio útil.
- El proxy tiene una referencia que apunta al objeto servicio. Cuando el proxy termina su procesado, pasa la petición al objeto servicio.
- 4. El cliente debe poder trabajar tanto con servicios como con proxys a través de la misma interfaz. Así se le puede pasar un proxy a un código que espera un servicio.





Caching results of a service with a proxy.

```
public listVideos()
{
    if (listCache == null)
    {
        listCache = service.listVideos();
    }
    return listCache;
}
```