

Modelado y programación  
Sección teórica: Practica 1

1- Principios de diseño: Strategy

Strategy se guía por principios como:

- Identificar los aspectos de que varían encapsulándolos y separarlos de los que no cambian.
- Programa a una interfaz, no a una implementación.
- Favorecer la composición sobre la herencia.

Principios de diseño: Observer

- Procurar que los objetos que interactúen entre sí estén débilmente acoplados
- Identificar los aspectos de que varían encapsulándolos y separarlos de los que no cambian.

Desventajas de: Strategy

- Se incrementa el número de clases que el programa debe mantener.
- En la implementación enseñada de strategy, se llega a ignorar el principio de programar a la interfaz.

Desventajas de: Observer

- Código más difícil de entender.
- Puede causar problemas en el manejo de la memoria, ya que sujeto va a mantener la referencia de un observador hasta que este cancela su registro.

README:

- En esta práctica se realizó una simulación de un chat para gente de España, Inglaterra y México, para hacer esta simulación más fácil de entender las acciones tal cual que vaya haciendo cada usuario (como mandar un mensaje, unirse al chat, etc.) se escriben con "--" al principio, y lo que supone que vería cada usuario en su pantalla se escribirá abajo de la acción y en el orden en el cual se fueron uniéndose al chat.
- Deje en el main una pequeña simulación del chat, se puede alterar utilizando los métodos `chat.setUltimoMensaje()` y `chat.remove()`, también creando nuevos usuarios.
- La práctica se ejecuta ingresando a la carpeta `src` desde la terminal y utilizando el comando `javac *.java` y luego `java Main` (Solo lo pongo por si acaso).