



# Práctica 01

## Modelado y Programacion 2022-2

"Paw patrols"  
Castellanos Palacios Gerardo 318245630  
Vargas Gutiérrez Julieta 318341945

Marzo 2022

## 1. Teoría

### 1.1. Strategy

Strategy es un patrón donde se encapsulan los comportamientos intercambiables de un algoritmo en una interfaz o clase abstracta, en las cuales se da una implementación de su comportamiento dependiendo lo que se quiere lograr.

- Principios de diseño
  1. Se presenta una abstracción en cada uno de los algoritmos los cuales son autónomos
  2. Su comportamiento es variable de programas y objetos
- Desventajas
  1. Debido a que se deben aplicar a todos los algoritmos se puede llegar a una sobredimension al tener tantos objetos en el programa

### 1.2. Observer

Observer es un patrón donde existe una dependencia de uno a muchos entre objetos, cuando el objeto cambiar su estado los que dependen en él serán notificados y actualizados.

- Principios de diseño
  1. Un observador depende de un sujeto
  2. Los observadores pueden ser agregados o borrados sin afectar el funcionamiento
- Desventajas
  1. En algunos casos puede resultar ineficiente notificar a todos los observables
  2. Si se desea excluir a algún observador de la notificación del cambio de estado no será posible ya que la notificación se envía a todos los observables

## 2. Práctica

### Descripción de la práctica

Debido al tema dado de los patrones de diseño *observer* y *strategy* se implementó una simulación la cual muestra a lo largo de un año el cobro de las empresas: Memeflix, Momazon, Sootify, HVO Max y Thisney a un cierto grupo de usuarios.

### Inconvenientes

Fue un tanto confuso poder diferenciar en qué parte de la simulación se usaría cada patrón de diseño, y por nuestras ideas erróneas tuvimos que cambiar el código y el uml múltiples veces.

### Compilación y Ejecución

Descomprimir la carpeta zip, colocarse en la carpeta src y en la consola ejecutar

```
javac Main.java
```

En la siguiente línea

```
java Main.java
```