### **Abstract Cliente**

# String nombre

Constructor(String nombre)

setNombre(String nombre)

getNombre():String

peliVista(Pelicula p):boolean

addValoracion(Pelicula p,int puntaje)

abstract getGenerosPref(): Vector<String>;

abstract getVistas(): Iterator<Pelicula>;

abstract addVista(Pelicula p);

### Usuario

int edad

Vector<String> generos

adds/set/removes

getGenerosPref():

Vector<String>

getVistas():

Iterator<Pelicula>

addVista(Pelicula p)

## Grupo

Vector<Cliente> integrantes

add/remove

getGenerosPref():Vector<String>

getVistas():Iterator<Pelicula>

addVista(Pelicula p)

## Pelicula

HashTable<Cliente,Integer> valoraciones

Vector<String> generos

Vector<String> directores

Vector<String> protagonistas

String titulo

int año

int duracion

Constructor(titulo,año)

set/get/add/remove

compararAños(int): int

contiteneGenero(String genero): boolean

contieneDirector(String director): boolean

contieneActor(String actor): boolean

putValoracion(Cliente cl,int puntaje)

equals(Object o): boolean

#### Sistema

int cantRecomendadas
Vector<Pelicula> peliculas
int puntajeMin
int puntajeMax

Constructor(int cantRec,int pMin, int pMax)
set/get/add/remove. //NO setPMin ni setPMax
buscar(Condicion c): Vector<Pelicula>
recomendar(Cliente cl, Comparator orden,
Condicion c, int cant): Vector<Pelicula>
recomendar(Cliente cl, Comparator orden,
Condicion c): Vector<Pelicula>
valorar(Cliente cl, Pelicula p, int puntaje)