

## Pràctica interfaces

### EXERCICI 1:

Es planteja desenvolupar un programa Java que permet representar la següent situació.

- Una *instal·lació esportiva* és un recinte delimitat a on es practiquen esports, en Java interessa disposar d'un mètode **int getTipusDeInstalacio()**.
- Un *edifici* és una construcció coberta i en Java interessa disposar d'un mètode **double getSuperficieEdifici()**.
- Un *poliesportiu* és al mateix temps una instal·lació esportiva i un edifici; en Java interessa conèixer la superfície que té i el nom que té.
- Un *edifici d'oficines* és un edifici; en Java interessa conèixer el nombre d'oficines que té.

Definir **dos interfaces** i **dos classes** que implementin una o dos interfaces (segons el cas) per a representar la situació anterior.

En una classe **test** amb el mètode **main**, crear un **ArrayList** que contengui tres poliesportius i dues edificis d'oficines i utilitzant un iterator, recórrer la col·lecció i mostrar els atributs de cada element.

### EXERCICI 2:

Crea una interface en Java que s'anomeni **InstrumentMusical** i que contingui els següents mètodes:

- tocar(): void
- afinar():void
- tipusInstrument(): String

Crea les classes **InstrumentVent** i **InstrumentCorda** que implementen la interface anterior. Crea la classe **exempleMusical** que conté els següents mètodes:

- **ManejoInstrument** que no retorna res i rep com a paràmetre un objecte de tipus instrumentMusical. Aquest mètode imprimeix el tipus d'instrument que és i crida als mètodes afinar i tocar.
- Mètode **main** en el que crearem els objectes guitarra i trompeta. Cridarem al mètode **ManejoInstrument** per a cadascun dels objectes creats.

### EXERCICI 3: Publicacions

Escriu un programa per a una biblioteca que contingui llibres i revistes.

- Els llibres i revistes són publicacions.
- Les característiques de les publicacions que s'emmagatzemen tant per a les revistes com per als llibres són el codi, el títol, i l'any de publicació. Aquestes tres característiques es passen per paràmetre en el moment de crear els objectes.
- Els llibres tenen a més un atribut prestat. Els llibres, quan es creen, no estan prestats.
- Les revistes tenen un nombre. En el moment de crear. En el moment de crear les revistes es passa el nombre per paràmetre.

- Tant les revistes com els llibres han de tenir (a part dels constructors) un mètode toString () que retorna el valor de tots els atributs en una cadena de caràcters. També tenen un mètode que retorna l'any de publicació, i un altre el codi.
- Per prevenir possibles canvis en el programa s'ha d'implementar una interfície prestat amb els mètodes prestar (), tornar () i prestat. La classe Llibre implementa aquesta interfície

#### EXERCICI 4: Cua del supermercat

---

- Escriu una interfície, anomenada ColeccioInterfaz, que declari els següents mètodes:
  - estaBuida (): retorna true si la col·lecció està buida i false en cas contrari.
  - extreure (): retorna i elimina el primer element de la col·lecció.
  - primer (): retorna el primer element de la col·lecció.
  - afegir (): afegeix un objecte per l'extrem que correspongui, i retorna true si s'ha afegit i false en cas contrari.
- Implementeu la classe Persona.
- A continuació, escriu una classe Pila, que implementi aquesta interfície, utilitzant com atributs un vector de Persones i un enter que serveix de comptador d'objectes.
- El constructor rep per paràmetre la mida màxima de la pila.
- El mètode estaBuida () comprova si el comptador és 0.
- El mètode afegir () comprova que cal l'element, i si és així, ho afegeix a la cel·la que indica el comptador, posteriorment s'incrementa el valor del comptador. Retorna true o false, segons si s'ha afegit o no.
- primer () si no està buit el vector, retorna l'element que està indicat pel comptador. Si està buit retorna null.
- extreure () si està buida, retorna null, si no decrementa el comptador i retorna l'element que està a la cel·la indicada pel comptador, després de decrementar-se.
- S'hauria de sobreescrivre el mètode toString ()