

## TP numéro 3

### Objectifs

- Tableaux ; Collections ; Interfaces et Gestion des Exceptions.

### Exercice 1

Écrivez un programme qui permet de créer le tableau suivant :

```
1 0 0 0
0 1 0 0
0 0 1 0
0 0 0 1
```

### Exercice 2

1. Écrire une classe **Etudiant** qui contient les attributs **id** (entier), **nom** et **prenom**.  
Le constructeur a un seul argument : **id**, et dans le constructeur on demande à l'utilisateur d'entrer le nom et le prenom.  
Ajouter une méthode **afficher()**.
2. Écrire une classe **GestionEtudiant** qui contient la méthode **main()**.
  - déclarer un `ArrayList` qui va contenir des étudiants ;

```
ArrayList<Etudiant> A = new ArrayList<Etudiant>();
```

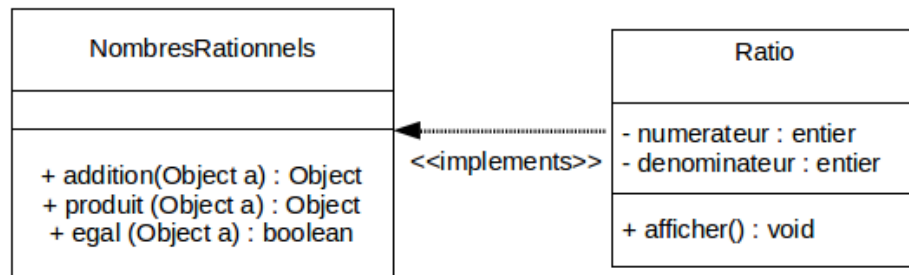
- écrire un menu qui contient : 1. **ajouter un étudiant** ; 2. **afficher les étudiants** ; 3. **quitter**.  
Le programme doit répondre à chaque cas.

### Exercice 3 (Extrait de l'examen normal 2019)

Un nombre rationnel est un nombre qui s'écrit sous la forme  $a/b$ , où  $a$  et  $b$  sont des entiers ( $b \neq 0$ ).

1. Écrivez une interface **NombresRationnels** qui contient les méthodes suivantes (voir figure) :
  - **addition()** qui a un seul argument de type `Object` et qui retourne un `Object`. Elle retourne la somme du nombre rationnel courant et celui passé en paramètre ;
  - **produit()** qui a un seul argument de type `Object` et qui retourne un `Object`. Elle retourne le produit du nombre rationnel courant et celui passé en paramètre ;
  - **egale()** qui a un seul argument de type `Object` et qui renvoie `true` si le nombre rationnel courant et celui passé en paramètre sont égaux (et renvoie `false` sinon).
2. Écrivez la classe **Ratio** qui implémente l'interface **NombresRationnels** et qui contient :
  - deux attributs **numérateur** et **denominateur** de type entier ;

- un constructeur qui a deux arguments. **Attention** : le programme lance une exception de type **ArithmeticException** si le denominateur est égal à 0 ;
  - une implémentation des trois méthodes : **addition()**, **produit()** et **egale()** ;
  - une méthode **afficher()** qui retourne une chaine de caractère de la forme  $a/b$ .
3. Écrivez une classe **Test** qui contient la méthode **main()**. Dans laquelle :
- déclarez et affichez les deux rationnels **Ratio(2,3)** et **Ratio(5,7)** ;
  - Appliquez les méthodes **addition()**, **produit()** et **egale()** à ces deux nombres, et affichez les résultats de chaque opération.
  - déclarez le nombre rationnel **Ratio(1,0)** ;



## Exercice 4 (Extrait de l'examen normal 2019)

1. Créez une classe **Vehicule** qui contient :
  - les attributs : **immatriculation**, **marque** et **couleur** (tous de type chaine de caractères) ;
  - un constructeur pour initialiser les différents attributs ;
  - une méthode **afficher()** qui affiche les trois attributs.
2. Créez une classe **Roue** qui est caractérisée par le type (chaine de caractères), la largeur (réel) et la hauteur (réel).
  - ajoutez un constructeur de trois arguments qui déclenche une **exception** si la largeur ou la hauteur est négative.
  - ajoutez une méthode **afficher()** qui affiche les informations d'une roue.
3. Créez une classe **Voiture** qui hérite de **Vehicule** et qui :
  - ajoute l'attribut **Roue** ;
  - redéfinit la méthode **afficher()**.
4. Créez une classe **Test** qui contient une méthode **main()**, dans laquelle :
  - déclarez une roue ;
  - instanciez une voiture qui contient la roue ainsi déclarée ;
  - affichez les informations de la voiture.