

## TD03 : Classes et objets

### Objectifs

- Savoir définir des classes ;
- Savoir instancier des classes ;
- Savoir envoyer des messages aux objets ;
- Comprendre l'utilisation du modificateurs d'accès `static` et `final` ;
- Savoir utiliser les variables et les méthodes de classe ;
- Savoir redéfinir et utiliser la méthode `toString` de la classe `Object` (classe mère de toutes les classes de Java) ;
- Savoir gérer des collections d'objets avec la classe `ArrayList` du package `Java.util`.

### Exercice 1. La classe « Etudiant »

Modéliser un étudiant par la classe `Etudiant` d'un package nommé `gestionEtudiants` de la façon suivante :

La classe `Etudiant` possède trois attributs privés :

- le prénom, nommé `prenom`, de type `String` ;
- le nom, nommé `nom`, de type `String` ;
- le matricule, nommé `matricule`, de type `int`.

La classe `Etudiant` possède deux constructeurs permettant respectivement d'initialiser :

- le prénom et le nom ;
- le prénom et le nom et le matricule.

La classe `Etudiant` possède aussi un *getter* et un *setter* pour chacun de ses attributs.

Enfin, la classe `Etudiant` possède une méthode `toString()` qui renvoie une chaîne de caractères décrivant l'étudiant.

### Exercice 2. La classe « Classe »

Modéliser une classe par la classe `Classe` du même package nommé `gestionEtudiants` de la façon suivante :

La classe `Classe` possède quatre attributs :

- L'année courante, nommé `annee`, de type `int` (Exemple : 2019) : cet attribut est une variable de classe publique et constante.
- La description de la classe, nommé `description`, de type `String` (Exemple : Master 1 MSIA) : cette attribut est une variable d'instance privée ;
- La liste des étudiants, nommé `etudiants`, de type `ArrayList<Etudiant>` : cette attribut est une variable d'instance privée ;
- Le nombre d'étudiants de la classe, nommé `nbrEtudiants`, de type `int` : cet attribut est une variable de classe publique.

La classe `Classe` possède un constructeur `Classe (String description)` permettant d'initialiser la description de la classe.

La classe `Classe` possède aussi un *getter* et un *setter* pour chacun de ses attributs.

La classe `Classe` possède une méthode `ajouterEtudiant (Etudiant etudiant)` qui ajoute un étudiant à la liste d'étudiants `etudiants`.

Enfin, classe `Classe` possède une méthode `afficherEtudiants ()` qui affiche la liste des étudiants de la classe.

### Exercice 3. La classe « Principale »

Tester les classes `Etudiant` et `Classe` en faisant au moins les opérations suivantes :

- Créer un premier objet de la classe `Etudiant` en utilisant le premier constructeur (celui qui initialise le prénom et le nom);
- Modifier le matricule de cet objet en utilisant le setter défini pour le matricule ;
- Créer deux autres objets de la classe `Etudiant` en utilisant le second constructeur (celui qui initialise le prénom, le nom et le matricule) ;
- Créer un objet de la classe `Classe` ;
- Ajouter les trois instances de la classe `Etudiants` que vous avez créées précédemment à liste des étudiants de l'instance de la classe `Classe` que vous venez de créer.
- Afficher la liste des étudiants.

Si les attributs `annee` et `nbrEtudiants` étaient déclarés privés, pourrions-nous y accéder dans la méthode `main` à partir du nom de la classe "`Classe`" ? Justifiez votre réponse.

Quelle serait la syntaxe correcte ?