

### Objectifs du TP :

- Implémentation correcte de l'évolution des objets dans le temps (héritage).
- Redéfinition des méthodes.
- Appel aux méthodes et constructeurs de la classe mère.

## Exercice 1

Une personne est caractérisée par son nom : String qui est déclaré private.

1. Définir la classe **Personne** avec en particulier une méthode d'instance (**int compare(Personne P)**) de comparaison suivant le nom.
2. Définir le constructeur **Personne(String nom)** qui crée et initialise un objet de type **Personne**.
3. Définir la méthode **String getNom()** qui retourne le nom d'une personne.
4. Définir la méthode **void setNom(String nom)** qui modifie le nom d'une personne.
5. Définir une méthode **sePresenter()** pour afficher *"Je suis une personne et je m'appelle <nom>"*

## Exercice 2

Un étudiant est caractérisé par son code : int, son université : String et sa moyenne.

1. Définir la classe **Etudiant** qui hérite de la classe **Personne**.
2. Définir le constructeur **Etudiant(String nom, int code, String universite, double moyenne)** qui crée et initialise un objet de type **Etudiant**.
3. Redéfinir la méthode **sePresenter()** pour afficher *"Je suis un étudiant à <université> et je m'appelle <nom>"*
4. Définir les méthodes **void setCode(int code)**, **void setUniversite(String universite)** et **setMoyenne(double moyenne)** qui modifient respectivement le code, l'université et la moyenne d'un étudiant.
5. Redéfinir la méthode **boolean equals(Object obj)** qui permet de comparer deux étudiants selon le code.
6. Dans le main, tester :
  - Une variable de type **Personne** qui contient un **Etudiant** (typage dynamique). Faire appel à la méthode **sePresenter()**.

## Exercice 3

Créer une classe **GroupeEtudiants** pour gérer un tableau d'étudiants.

Ajouter :

1. Une méthode pour rechercher un étudiant par son code.
2. Une méthode pour ajouter un étudiant au tableau. On suppose que le nombre maximal d'étudiants est 100.
3. une méthode pour modifier le nom d'un étudiant.
4. Une méthode pour afficher tous les étudiants.
5. Une méthode pour calculer la moyenne générale des étudiants.
6. Un menu de choix pour tester les appels aux méthodes (Utilisation de switch case).