



## *Cours de Java*

### ***Projet 4 : Collections***

#### **Modalités :**

- Travail en autonomie
- Production individuelle

#### **Objectifs de l'activité :**

- Manipuler les collections Java
- Validation des compétences n°6 et n°13

#### **Consignes :**

##### **1. Commencer par étudier ces tutoriels proposés :**

- OpenClassrooms : « Collections »
- CodeCademy : « Data Structures »

On va maintenant gérer une liste d'utilisateurs, créer une classe Message (expéditeur, destinataire, contenu...) et la rajouter dans l'historique des messages (collection de tous les messages), et pour chaque utilisateur, gérer une collection d'amis.

##### **2. Ecrire le programme final :**



- Vous allez choisir entre vous connecter (si vous êtes déjà dans la liste des personnes), vous inscrire (pour vous rajouter à la liste des personnes), afficher la liste des personnes, ou quitter le programme.
  - Si vous vous connectez, vous allez saisir votre nom et votre prénom et si vous êtes déjà dans la collection, afficher « Vous êtes bien connecté »
  - Si vous vous inscrivez, vous allez saisir les infos (qui vont dépendre de votre rôle) et vous rajouter à la collection des utilisateurs et afficher « vous êtes bien connecté »
  - Dans ces deux cas :
  - Si vous êtes utilisateur on va ensuite
    - Afficher la liste de ses amis,
    - Se rajouter un nouvel ami (chercher dans la collection des utilisateurs, rajouter l'ami dans notre collection d'amis)
    - Supprimer un ami (de notre collection d'ami)
    - Rechercher si « toto » est notre ami
    - Rechercher si « toto » est inscrit (collection d'utilisateurs)
    - Envoyer un nouveau Message, et le rajouter dans l'historique de tous les messages.
    - Afficher la liste des messages envoyés
    - Afficher la liste des messages reçus
  - si vous êtes modérateur :
    - idem précédemment
    - Supprimer un message de l'historique
  - si vous êtes directeur :
    - implémenter la méthode « rajouterPersonne » en créant un employé.

### Livrables :

- Code des classes demandées. A déposer sous git.
- Diagramme de Use Case correspondant aux fonctionnalités décrites
- Glossaire enrichi des différentes syntaxes utilisées. A déposer sous le nom ***glossaire***, dans votre répertoire individuel sous ***java → livrables***

### Ressources :

- CodeCademy <https://www.codecademy.com/en/courses/learn-java/lessons/data-structures/exercises/data-structures>
- Openclassrooms <https://openclassrooms.com/courses/apprenez-a-programmer-en-java/les-collections-d-objets>
- Tutoriel d'Oracle pour Java <http://docs.oracle.com/javase/tutorial/java/index.html>

## Synthèse (activité de groupe)

### Modalités :

- Travail en classe mixée

## **Objectifs de l'activité :**

- Activité de synthèse de type recherche d'erreurs.

## **Déroulement :**

- Slides fournis pour l'activité de synthèse
- Recherche individuelle puis mise en commun