

# IS2A4 – PPO – T.P. 6a

© Polytech Lille

## Résumé

Ce TP aborde la sérialisation en java.

N'hésitez pas à consulter la documentation en ligne sur le site d'oracle : <https://docs.oracle.com/javase/8/docs/api/> (regardez la version correspondant la version de java que vous utilisez).

## I Préparation du TP (5 min)

- I.1. Créer un dossier spécifique pour les TP de PPO qui vous appellerez TP\_PPO par exemple.
- I.2. Dans ce dossier, créer un dossier TP6a. C'est dans ce dossier que vous travaillerez aujourd'hui. Durant les prochaines séances de TP, essayez de garder cette pratique.
- I.3. Récupérez le fichier TP6a\_source.zip sur moodle et le décompresser dans TP6a.

## II Write et Read (40 min)

L'objectif de cette section est de créer deux classes `Write` et `Read` qui pour l'un écrira dans un fichier une bibliothèque et dans l'autre récupérera l'information pour l'afficher.

- II.1. Modifiez votre classe `Auteur` pour la rendre sérialisable (il vous suffit juste de souscrire ou contrat de l'interface `Serializable`).
- II.2. Modifiez votre classe `Livre` pour la rendre sérialisable. Comme les enfants héritent des contrats de leurs classes parentes, les classes `OuvrageColl` et `OuvrageMono` sont alors automatiquement sérialisable.
- II.3. Créez deux méthodes `save` et `load` dans la classe `Bibliotheque` :
  - la méthode `save` prendra comme entrée un nom de fichier et écrira votre liste de livres dedans.
  - la méthode `load` prendra comme entrée un nom de fichier et récupérera votre liste de livres dedans.
- II.4. Créez deux classes `Write` et `Read` :
  - la classe `Write` créera une bibliothèque et la sauvegardera dans un fichier (par exemple `save.bin`).
  - la classe `Read` chargera une bibliothèque à partir de ce même fichier.
- II.5. Testez votre code (n'oubliez pas de gérer les exceptions `IOException` et `ClassNotFoundException`).
- II.6. Modifiez votre classe `Bibliotheque` en ajoutant un constructeur qui prend un nom de fichier et crée la bibliothèque sauvegardée dans ce fichier.
- II.7. Retestez votre code (n'oubliez pas de gérer les exceptions `IOException` et `ClassNotFoundException`).

## III Lire un fichier CSV (25 min)

L'objectif de cette section est de transférer un fichier CSV en un fichier pouvant être lu par votre classe `Read`.

- III.1. Lisez et testez le fichier de classe `LireCSVAvecScanner`.
- III.2. Créez une classe `WriteCSV` qui lira un fichier CSV et ajoutera un livre dans la bibliothèque pour chaque ligne dans le fichier.
- III.3. Testez votre code en essayant de lire votre fichier à l'aide de la classe `Read`.
- III.4. Modifiez vos classes `WriteCSV` et `Read` pour que le nom des fichiers ne soient plus écrit en dure dans le fichier mais passés en paramètre.

Temps total du sujet : 1 heures et 10 minutes