

Programmation avancée — Java

TP7 – Générateur multi-threadé

1 Introduction

Les threads sont couramment utilisés dans le développement. Ils servent à paralléliser les traitements pour gagner en vitesse d'exécution et rendre plus réactifs les programmes. C'est donc une notion fondamentale à maîtriser pour tout développeur.

Ce TP consistera à créer un générateur automatique et multi-threadé de données. Nous allons illustrer cela avec des étudiants. Ces étudiants vont posséder plusieurs champs : identifiant national étudiant, nom, prénom, date de naissance, lieu de naissance et un petit descriptif de leur parcours.

La seule documentation autorisée est la documentation officielle Java <http://docs.oracle.com/javase/8/docs/api/> (pour le reste **vous me demandez**).

2 Travail à effectuer

1. Créez la classe Etudiant avec ses différents champs
2. Créez un générateur retournant un étudiant créé avec des données aléatoires.
3. Créez un programme séquentiel qui génère une liste de 150000 étudiants.
4. Mesurez le temps d'exécution de votre programme
5. Modifiez votre programme pour qu'il crée 3 Thread qui créent chacun 50000 étudiants. N'oubliez pas que vous voulez une liste de 150000 étudiants et pas 3 listes de 50000.
6. Mesurez le temps d'exécution. Combien avez-vous gagné? Si vous ne gagnez pas en performances (ou pas assez à votre goût), analysez votre gestion de la liste d'étudiants entre les Threads et voyez comment améliorer cela.

Bonus Sauvegardez les 150000 étudiants sur le disque le plus rapidement possible. Mesurez l'impact sur les performances.

Note : pour mesurer les performances d'un programme, il faut généralement le lancer à de multiples reprises puis faire une moyenne du temps d'exécution

3 Livraison

Ce TP doit être rendu terminé avant le 23 novembre à 23h59.

Le transmettre à o.goutet@openium.fr dans un fichier compressé (tar gz ou bz2) avec votre promotion, filière, nom et numéro de TP. Exemple : 2021_F5_pateyron_TP7.tgz.