

## Activité pratique : Les threads

### Exercice 1 :

On souhaite créer une simple application pour mettant en évidence un comportement classique en programmation multi-threads.

1. Écrivez une classe **Talkative** qui contient un constructeur prenant en paramètre un entier qui est un attribut de la classe ;
2. Modifiez votre classe afin qu'elle implémente **Runnable** ;
3. Redéfinissez la méthode **run** afin qu'elle affiche 100 fois la valeur de l'attribut contenu par la classe ;
4. Dans une méthode statique **main** créez 10 instances de la classe **Thread**. Chacune de ces instances prendra en paramètre une nouvelle instance d'un objet de type **Talkative**. Chacun des objets de type **Talkative** doit être construit en prenant en paramètre un entier unique ;
5. Appelez la méthode **start** sur chacun des objets de type **Thread** qui ont été créés.
6. Que constatez-vous ?

### Exercice 2 :

On souhaite créer un programme Java qui utilise un pool de threads pour calculer la somme des éléments d'un tableau de manière parallèle. Pour cela vous avez les renseignements suivants :

- Créez une classe **Sommeur** qui implémente l'interface **Runnable**.
- Le constructeur de la classe **Sommeur** prendra comme argument un tableau d'entiers et deux indices **debut** et **fin** pour définir la plage d'indices sur laquelle la somme doit être calculée.
- La classe **Sommeur** doit avoir une méthode **getSomme()** qui renvoie la somme calculée dans la plage spécifiée.
- Dans la classe principale (**Main**), créez un tableau d'entiers.

- Divisez le tableau en plusieurs plages et créez un pool de threads pour calculer la somme de chaque plage de manière concurrente.
- Attendez que tous les threads aient terminé et affichez la somme totale du tableau.