

Programmation parallèle et concurrente en Java



En programmation Web, avec Java EE ou Spring Framework, l'essentiel de l'aspect multi-tâches est caché au développeur. Mais lorsqu'il s'agit de développer des batchs ou des traitements massifs, il faut souvent revenir aux racines des Threads et des fonctionnalités du JDK.

Dans cette formation, on commence pas manipuler les Threads de façon brute, comme en 1999, puis on voit les différentes classes et possibilités offertes par le JDK pour faciliter le développement.

2 jours

Tarif

- intra : 2560 euros HT
(maxi 6 participants)

- inter : 1160 euros HT

Introduction

- ▶ Les bénéfices des traitements parallèles
- ▶ Les risques liés aux accès concurrents

Premiers pas

- ▶ La classe Thread
- ▶ Les tâches Runnable
- ▶ Les techniques de blocage de thread
- ▶ Comment interrompre un thread ?
- ▶ La gestion des erreurs

Gestion moderne des threads

- ▶ Le framework Executor
- ▶ Les pools de threads
- ▶ Les tâches Callable, le Future
- ▶ L'interruption de Threads gérés
- ▶ Le pool de Fork / Join

Accès concurrents

- ▶ Problèmes d'accès concurrent aux données et ressources
- ▶ Le dead lock, leur détection
- ▶ Les objets atomiques
- ▶ Les techniques avancées de blocage : barrières et loquets
- ▶ Les collections concurrentes

Nouveautés Java 8

- ▶ L'utilisation de Lambda avec les classes et interfaces existantes
- ▶ Les traitements parallèles avec les tableaux
- ▶ Les collections et les streams parallèles
- ▶ La programmation de style *reactive*, avec `CompletableFuture`
- ▶ Quelques autres nouveautés : classes et méthodes