

# Formation : Java, optimisation : CPU, mémoire

Développer des applications efficaces et fiables

Formation pratique - 4j - 28h00 - Réf. JOT

Prix : 2470 CHF H.T.

★★★★☆ 4,2 / 5

Optimiser les performances des applications Java, en respectant certains patterns de programmation. Analyser certains mécanismes de la JVM qui contribuent à l'obtention de performances ainsi que les aspects spécifiques au multithread. Cette formation met l'accent sur les aspects CPU et gestion de la mémoire.

## Objectifs pédagogiques

À l'issue de la formation, le participant sera en mesure de :

- ✓ Optimiser le code Java d'une application pour en améliorer les performances
- ✓ Maîtriser les outils d'aide à l'optimisation
- ✓ Optimiser le fonctionnement de la machine virtuelle
- ✓ Optimiser l'utilisation des bibliothèques standards
- ✓ Optimiser l'usage du multithreading

## Public concerné

Développeurs, ingénieurs, chefs de projets proches du développement.

## Prérequis

Bonnes connaissances du langage Java. Expérience requise en programmation Java.

Vérifiez que vous avez les prérequis nécessaires pour profiter pleinement de cette formation en faisant [ce test](#).

## Méthodes et moyens pédagogiques

### Travaux pratiques

Les TP de cette formation ne comportent pas de programmation mais consistent en l'utilisation d'outils et en analyse de code optimisé.

### Méthodes pédagogiques

Vous vous familiariserez durant toute la session avec des outils d'aide à l'optimisation.

### PARTICIPANTS

Développeurs, ingénieurs, chefs de projets proches du développement.

### PRÉREQUIS

Bonnes connaissances du langage Java. Expérience requise en programmation Java.

### COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

### MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques...

Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

## Modalités d'évaluation

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques...

Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

## Programme de la formation

### 1 Méthodologie et outils pour l'optimisation

- Les principes, les stratégies d'optimisation.
- Les outils de profilage, de tests et d'aide.
- Les éléments impliqués dans l'optimisation.

#### Travaux pratiques

Utilisation d'un outil de profilage (VisualVM ou JFR) et d'un outil de tests unitaires (JUnit).

### 2 Les caches

- Le principe général des caches, les méthodes importantes.
- Les principales collections utilisées comme caches.

#### Travaux pratiques

L'amélioration des performances de l'application témoin par l'introduction de caches. Utilisation d'un outil de profilage (VisualVM ou JFR).

### 3 L'optimisation et la gestion de la mémoire

- La création, la copie d'objets, de tableaux (clone vs constructeurs de copie).
- Les objets non modifiables, les objets non mutables, les pools d'objets.
- Les différents GC (les algorithmes, le paramétrage, la visualisation de l'activité).
- Les dumps mémoire, la saturation mémoire (les causes, le diagnostic, les solutions).
- Les références faibles (soft, weak, phantom);

#### Travaux pratiques

L'amélioration des performances de l'application témoin par une meilleure gestion des objets. Utilisation d'un outil de visualisation de la mémoire (VisualVM) et de l'activité du GC (Visual GC).

### 4 L'optimisation et le multithread

- Le parallélisme versus la concurrence.
- Les principaux concepts : Runnable, Thread, Callable<T>, ExecutorService, ...
- L'impact du parallélisme et de la concurrence sur les performances.
- Quelques solutions d'optimisation.

#### Travaux pratiques

L'amélioration des performances de l'application témoin par l'introduction de threads (la parallélisation et la gestion de la concurrence).

## MOYENS PÉDAGOGIQUES ET TECHNIQUES

- Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les formations pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
- À l'issue de chaque formation ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
- Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le participant a bien assisté à la totalité de la session.

## MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

## ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Pour toute question ou besoin relatif à l'accessibilité, vous pouvez joindre notre équipe PSH par e-mail à l'adresse [psh-accueil@orsys.fr](mailto:psh-accueil@orsys.fr).

## 5 L'optimisation et le langage

- Les différents modes d'exécution et de compilation.
- L'utilisation optimale des types, des instructions, des méthodes.
- Les lambda-expressions, les streams, ...
- La manipulation des chaînes de caractères.

### Travaux pratiques

Amélioration des performances de l'application témoin.

## 6 L'optimisation et les I/O

- La librairie nio. Les buffers, channels. Les sélecteurs et leur utilisation.
- Le choix des bonnes classes de stream.
- L'optimisation de la sérialisation.

### Travaux pratiques

Amélioration des performances de l'application témoin par une meilleure gestion des IO.

## Dates et lieux

### CLASSE À DISTANCE

2026 : 23 juin, 3 nov.