

# Formation Java, Programmation avancée

## Présentation

Cette formation avancée de 4 jours fait passer les développeurs Java à un niveau expert : elle consolide les nouveautés du langage, introduit la programmation concurrente sécurisée et le chargement dynamique de classes, approfondit l'accès aux données et les tests modernes, puis met en pratique les principaux frameworks (Spring Boot, Apache Commons, Guava) au travers d'un mini-projet REST conteneurisé. Un volet « IA légère » montre comment exploiter un LLM pour suggérer du code et prioriser les tickets, illustrant concrètement l'intégration de l'intelligence artificielle dans les workflows Java.

Durée : 28,00 heures (4 jours)

Tarif INTRA : [Nous consulter](#)

## Objectifs de la formation

A l'issue de la formation, le stagiaire sera capable d'utiliser les fonctions avancées du langage Java dans les projets de développement d'applications.

- Maîtriser les aspects avancés du langage Java
- Comprendre comment appliquer les principaux frameworks et bibliothèques Java.
- Comprendre le modèle de sécurité et le chargement des classes Java
- Savoir programmer en Java dans le contexte des bases de données relationnelles
- Savoir tester, debugger et optimiser ses applications
- Appréhender les nouveautés Java

## Prérequis

Maîtriser le langage Java, connaître les concepts de bases de données relationnelles et du langage SQL, disposer d'une expérience en programmation Java

## Public

Développeurs, chargés de développement d'applications informatiques, chefs de projets proches du développement...



## Programme de la formation

### Jour 1 – Aspects avancés du langage Java

Objectifs :

- Consolider les bases avancées Java
- Maîtriser les nouveautés du langage Java
- Appliquer la généricité et les énumérations avancées

### Rappels et nouveautés Java

- Nouveautés Java : Streams, Lambdas et références de méthodes (Java 8)
- Généricité avancée (bornes, wildcards) et énumérations enrichies
- Inférence de type (Java 10) et JShell (Java 9)
- Introduction aux principes SOLID

### Atelier pratique

- Exercices de refactoring pour intégrer Lambdas, Streams, Records dans un code existant
- Détecter des violations SOLID

### Jour 2 – Concurrency, sécurité du chargement des classes, design patterns

Objectifs :

- Comprendre la programmation concurrente et le modèle de threads Java
- Maîtriser le chargement dynamique et la sécurité des classes

### Threads et concurrence

- Introduction à Thread, Runnable et API Concurrency
- Multithreads : synchronisation, attentes et gestion des priorités
- Threads virtuels (Java 17)

### Design patterns

- Singleton, Factory, Strategy, Observer
- Pattern plugin pour le chargement dynamique des classes

### **Modèle de sécurité et ClassLoader**

- Architecture du ClassLoader
- Sécurité de chargement : sandboxing et Access Controller
- Policy files

### **Atelier pratique**

- Charger un traitement asynchrone avec des threads virtuels
- Charger dynamiquement des classes et gérer les permissions

### **Jour 3 – Bases de données, tests et IA**

Objectifs :

- Programmer en Java avec JDBC et JPA
- Tester, debugger et optimiser les applications
- Découvrir l'usage de l'IA légère pour l'aide au développement

### **Accès aux données**

- JDBC : drivers, DAO et pattern singleton
- Introduction à JPA/Hibernate : mapping, transactions et validations

### **Tests, debug et optimisation**

- JUnit avancé, fixtures et mocking
- BDD simple avec Gherkin (Cucumber/Behave)
- Profiling et tuning de performance

### **IA pour développeurs**

- Prompt engineering pour suggérer des tests unitaires et des corrections de code

### **Atelier pratique**

- Développer un service persistant (JPA)

- Créer deux tests puis intégrer une API de génération de code pour améliorer les tests

## **Jour 4 – Frameworks, librairies et nouveautés Java**

Objectifs :

- Appliquer les principaux frameworks et librairies Java
- Découvrir les dernières évolutions de la plateforme Java

### **Frameworks et librairies**

- Spring Boot : concepts clés et micro-services
- Bibliothèques courantes : Apache Commons, Guava, Log4j

### **Nouveautés récentes**

- Modules (Java 11), HTTP Client (Java 11+)
- Coûts et bénéfices des fonctionnalités latest LTS

### **Introduction CI/CD**

- Github Actions

## **Atelier pratique de fin de formation**

Scénario : créer un service REST sécurisé en Java avec Spring Boot pour gérer des tickets de support

Tâches :

1. Définir une entité JPA Ticket et un repository Hibernate
2. Implémenter un contrôleur REST pour les opérations CRUD avec validation des droits (JWT)
3. Intégrer un appel IA : utiliser un LLM pour analyser le champ description et proposer une priorisation automatique
4. Rédiger des tests unitaires (JUnit + Mockito) et un scénario BDD minimal (Cucumber)
5. Construire un Dockerfile et exécuter l'application dans un conteneur

## Organisation

### Formateur

Les formateurs de Docaposte Institute sont des experts de leur domaine, disposant d'une expérience terrain qu'ils enrichissent continuellement. Leurs connaissances techniques et pédagogiques sont rigoureusement validées en amont par nos référents internes. Riches de leur expérience sur le sujet, ils sauront accompagner vos collaborateurs dans leur montée en compétences.

### Moyens pédagogiques et techniques

- Apports des connaissances communes.
- Mises en situation sur le thème de la formation et des cas concrets.
- Méthodologie d'apprentissage attractive, interactive et participative.
- Équilibre théorie / pratique : 60 % / 40 %.
- Supports de cours fournis au format papier et/ou numérique.
- Ressources documentaires en ligne et références mises à disposition par le formateur.
- Pour les formations en présentiel dans les locaux mis à disposition, les apprenants sont accueillis dans une salle de cours équipée d'un réseau Wi-Fi, d'un tableau blanc ou paperboard. Un ordinateur avec les logiciels appropriés est mis à disposition (le cas échéant).

### Dispositif de suivi de l'exécution et de l'évaluation des résultats de la formation

En amont de la formation :

- Recueil des besoins des apprenants afin de disposer des informations essentielles au bon déroulement de la formation (profil, niveau, attentes particulières...).
- Auto-positionnement des apprenants afin de mesurer le niveau de départ.

Tout au long de la formation :

- Évaluation continue des acquis avec des questions orales, des exercices, des QCM, des cas pratiques ou mises en situation...

A la fin de la formation :

- Auto-positionnement des apprenants afin de mesurer l'acquisition des compétences.

- Évaluation par le formateur des compétences acquises par les apprenants.
- Questionnaire de satisfaction à chaud afin de recueillir la satisfaction des apprenants à l'issue de la formation.
- Questionnaire de satisfaction à froid afin d'évaluer les apports ancrés de la formation et leurs mises en application au quotidien.

### **Accessibilité**

Nos formations peuvent être adaptées à certaines conditions de handicap. Nous contacter pour toute information et demande spécifique.