

## Java 8

Assimiler et mettre en œuvre la version 8 de Java

La version 8 de Java SE est probablement la version la plus importante proposée depuis sa création.

Java 8 apporte de nombreuses nouveautés au niveau du langage (Expression lambda, les méthodes par défaut, les références de méthodes, ...), de nouvelles API (Date & Time, Nashorn,... ), de nouvelles fonctionnalités (concaténation de chaînes, traitements en parallèle sur des tableaux, concurrency, Base64, ...) et des évolutions dans la JVM.

Elle introduit un nouveau modèle de programmation reposant sur une approche fonctionnelle au travers de l'API Stream.

Ce cours propose une vue d'ensemble détaillée de ces fonctionnalités qui changent la manière de concevoir et développer des applications.

Cette formation est le prérequis à la formation Java 9,10,11 !!

### Détails

- Code : JA-J8
- Durée : 2 jours ( 14 heures )

#### Public

- Developers
- Développeur web
- Développeurs
- Développeurs
- Développeurs Android
- Développeurs d'application
- Développeurs expérimentés
- Développeurs Java
- Directeur Technique
- Directeurs de projets
- Experienced object-oriented developers

#### Pré-requis

- Connaissance pratique d'une version antérieure de Java

#### Objectifs

- Assimiler la version 8 de Java

### Programme

#### Introduction

- JCP, JSR, JEP
- Historique de Java
- Les nouveautés de Java 8

#### Les interfaces

- Les méthodes par défaut et static

#### Les classes Value-Based

#### Les expressions Lambda

- La syntaxe
- Les interfaces fonctionnelles
- Le package java.util.Function
- Les références de méthodes
- La capture de variables

#### La classe Optional

#### L'API Stream

- Obtenir un Stream
- Les opérations
- Les opérations intermédiaires
- Les opérations terminales
- La classe Collectors
- Les Stream parallèles

#### Date and Time API

- L'API Date & Time
- Des classes dédiées
- Des opérations consistantes
- La représentation machine
- La représentation humaine
- Les classes TimeZone et Offset
- Des constantes et énumérations
- L'interface TemporalAdjuster
- La classe abstraite Clock

- Le formatage et le parsing
- L'intégration avec le code existant

### Collection

- Comparator
- Iterable.forEach()
- Collection.removeIf()
- List.replaceAll()
- List.sort()
- Map

### Parallel arrays

- Les opérations en parallèle sur des tableaux
- Initialiser les éléments
- Trier les éléments

### Concurrency

- Les ajouts dans Java 8

- Adder et Accumulator
- StampedLock
- ConcurrentHashMap

### La concaténation de chaînes

### Les annotations

- Java Type Annotations
- Repeating Annotations

### Base64

### Nashorn

### Reflexion

- Accès aux noms des paramètres

### JVM

- Metaspace

## Modalités

- **Type d'action** :Acquisition des connaissances
- **Moyens de la formation** :Formation présentielle – 1 poste par stagiaire – 1 vidéo projecteur – Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** :Exposés – Cas pratiques – Synthèse
- **Validation** :Exercices de validation – Attestation de stages