

# Java Perfectionnement

5 jours  
35 heures

log-081.pdf



loging-formation.com

## Objectifs

**Approfondissement des connaissances en découvrant les aspects avancés du Java. Réaliser des applications graphiques en respectant les architectures recommandées pour ce genre de programmes. Découverte des problématiques de la sécurité et de l'accès au réseau.**

## Participants

**Programmeurs, chefs de projets.**

## Prérequis

**Avoir les notions de la formation Java les bases.**

## Pédagogie

**La pédagogie est basée sur le principe de la dynamique de groupe avec alternance d'apports théoriques, de phases de réflexion collectives et individuelles, d'exercices, d'études de cas et de mises en situations observées. Formation / Action participative et interactive : les participants sont acteurs de leur formation notamment lors des mises en situation car ils s'appuient sur leurs connaissances, les expériences et mettront en oeuvre les nouveaux outils présentés au cours de la session.**

## Profil de l'intervenant

**Consultant-formateur expert sur cette thématique. Suivi des compétences techniques et pédagogiques assurée par nos services.**

## Moyens techniques

**Encadrement complet des stagiaires durant la formation. Espace d'accueil, configuration technique des salles et matériel pédagogique dédié pour les formations en centre. Remise d'une documentation pédagogique papier ou numérique à échéance de la formation.**

## Méthodes d'évaluation des acquis

**Exercices individuels et collectifs durant la formation. Evaluation des acquis et attestation de fin de stage adressés avec la facture.**

## Programme

### Concepts et syntaxes avancés du langage Java

**Les « imports static »**

**Autoboxing et auto-unboxing des types primitifs**

**Les types génériques**

**Amélioration des boucles avec les collections**

**Méthodes à nombre d'arguments variable**

**Annotations : utilisation, création**

**Gestion des interfaces**

**Gestion de la mémoire : GC**

## Java Perfectionnement

5 jours  
35 heures

log-081.pdf



loging-formation.com

### Gérer les accès concurrents en Java

Définition d'un thread

Créer et manipuler les threads

Synchroniser les threads sur un objet moniteur

Synchronisation : wait, notify, mutex, ....

Gestion du piping : stream

Bonnes et mauvaises pratiques

### Flux d'entrée/Sortie (I/O) en Java

Gestion des entrées/sorties,

Les API Java de gestion des flux,

Programmation par exception,

Sérialisation / Dé sérialisation.

### Principes des architectures n-tiers, pattern MVC et Design Patterns

Le couple Observer/Observable

Façade

Fabrique abstraite

Inversion de contrôle

Itérateur et Visiteur

### Accéder à une base de données relationnelle avec JDBC

Architecture et types de pilotes JDBC

Charger le pilote et récupérer une connexion à la base de données,

Exécuter des requêtes SQL,

Requêtes paramétrées,

API RowSet

Utilisation de procédures stockées

Gestion des transactions

Persistance et ORM

### Gestion de flux d'entrées/sorties en Java

Les classes InputStream/OutputStream

Reader/Writer ou entrées/sorties en mode caractères

Manipulation du système de fichiers

## Java Perfectionnement

5 jours  
35 heures

log-081.pdf



[loging-formation.com](http://loging-formation.com)

**NIO 2, channels, opérations asynchrones**