

JAVA, LANGAGE ET OUTILS

Objectifs <ul style="list-style-type: none"> ✓ Acquérir les bases de Java ✓ Avoir une vision d'ensemble de J2EE ✓ Maîtriser les outils ✓ Voir les infrastructures annexes (osgi, spring...) ✓ Connaître les bonnes pratiques 	Public <ul style="list-style-type: none"> ✓ Développeur ✓ Chef de projet 	Prérequis <ul style="list-style-type: none"> ✓ Connaissance de la programmation 	 <p>Durée 5 jours</p>
--	---	---	--

Introduction

- Présentation de Java, J2EE
- Termes et notions de base

Notions de conception

- Interfaces / Classes / Packages
- Héritage / Polymorphisme
- Composition / Association
- Bonnes pratiques

Langage Java

- Syntaxe de base, de conception et de programmation
- Tableaux, dates, chaînes
- Organisation et lancement d'un projet

Définition d'une classe

- Contenu, règles à suivre
- Structure interne
- Méthodes, champs et modificateurs
- Héritage, Implémentation d'interfaces.
- Utilisation des generics

Autres concepts

- Gestion des exceptions, erreurs, runtimeExceptions
- Description des différentes collections
- Utilisation des itérateurs
- Gestion des Threads, partage de ressources, blocages
- Utilisation des threads dans les interfaces
- Entrées/Sorties, fichiers, streams, sérialisation d'objets

IHM

- Présentation AWT / Swing
- Les différents composants
- La gestion des événements
- Comparaison avec SWT/JFace

Mémoire

- Gestion de la mémoire
- Paramétrage au lancement
- Suivi et contrôle

Industrialisation

- Livrer son application / Packaging
- Ant, outil d'aide au build
- Java Web Start

Autres Notions

- Accès aux Bases de données, JDBC
- Les servlets
- Distribution d'objets en RMI
- Lien avec C/C++ avec JNI

Infrastructures complémentaires

- OSGi, Spring, Spring DM, Eclipse RCP
- Présentation, utilisation, interactions

Eclipse

- Eclipse en tant qu'outil
- Projet java, paramétrage, gestion
- Lancement de projet
- Environnement de développement
- Navigation
- Mises à jour, outils additionnels
- Trucs et Astuces