



Meilleures pratiques Java Jee

Plan de cours détaillé

Objectifs:

Développer les compétences clés et les réflexes afin d'assurer un refactoring d'une application Java existante, ou de mieux programmer en Java.

Connaissances préalables :

Une expérience pratique dans le développement java est indispensable. Voir notre formation OBJFO1

Profils participants :

développeurs, chefs de projet

Durée : 4 jours

Outils éprouvés pour meilleure productivité du développeur <ul style="list-style-type: none">✦ Automatisation du processus de génération en utilisant Ant✦ Contrôle et configuration de la journalisation avec Log4J✦ Tests unitaires des composants complexes✦ Constituer et maintenir les tests JUnit✦ Automatisation des tests sur l'intégralité du projet✦ Tests de servlets, JSP, EJB	Refactoring et amélioration de la qualité du code <ul style="list-style-type: none">✦ Élimination des erreurs d'exécution grâce aux types génériques✦ Limitation des valeurs de paramètre avec la canonicalisation✦ Élargir l'applicabilité avec l'introspection✦ Simplifier l'introspection avec les JavaBeans et les annotations✦ Simplification du code source avec la refactorisation✦ Conception d'interfaces pour une meilleure flexibilité logicielle✦ Design patterns orientés objet clés : Factory, Composite, IoC, Stratégie✦ Automatisation contrôle erreurs courantes
Performances <ul style="list-style-type: none">✦ Outils d'analyse des performances✦ Évaluation du temps de réponse✦ Techniques de gestion des problèmes de performances courants de Java✦ Utilisation du ramasse-miettes✦ Choix des paramètres adaptés pour la JVM et le conteneur✦ Évaluation des besoins en NIO et JNI✦ Réorganisation des boucles pour améliorer le temps de réponse✦ Synchronisation des processus légers✦ Techniques de partage de données entre les processus	Sécurisation des applications <ul style="list-style-type: none">✦ Codage sécurisé en Java✦ Restriction d'accès aux ressources protégées✦ Établissement de règles de sécurité✦ Application de la sécurité basée sur les rôles✦ Authentification des utilisateurs dans des applications Web
Programmation orientée aspects <ul style="list-style-type: none">✦ Inversion de contrôle par les Beans Factories✦ Injection de comportement avec les aspects✦ Doter une application de capacités de scripting✦ Évaluation sécurisée de scripts d'utilisateurs finaux	Architecture applicative <ul style="list-style-type: none">✦ Découpage en couches✦ programmation par interface✦ Comparaison frameworks de vue✦ Comparaison frameworks de persistance