

Programmation Java : les bonnes pratiques de développement avec Rational Application Developer (RAD)

Cette formation constitue un point d'entrée pour toutes les personnes qui souhaitent s'orienter vers le développement Java.

Java est à la fois un langage de programmation et une plate-forme d'exécution. Il est la pierre de base du développement d'applications basées sur la technologie Java EE (WebSphere, WebLogic, JBoss, Tomcat, AppServer...).

Programme

Architecture de la plate-forme Java

- Les caractéristiques du langage
- La machine virtuelle
- Les outils standards
- Projet, compilation, exécution, débogage, génération de documentations
- Conception de scripts d'exploitation avec Ant (compilation, déploiement,...)
- Les évolutions et les versions

Mise en oeuvre de RAD

- Les concepts de RAD
- L'environnement, les perspectives, les vues, les plugins
- Compiler, exécuter et mettre au point un programme Java
- Exemple de modèle UML et démonstration de génération de code Java

Les éléments syntaxiques du langage et les packages de base

- Structure d'une application
- Les types primitifs, les classes
- Les structures de contrôle
- Mise en oeuvre des concepts objet
- Classe, attributs et méthodes
- Encapsulation, intégrité, évolutivité
- Les tableaux
- Les packages
- Mise en oeuvre de l'héritage
- Polymorphisme
- Les interfaces
- Les collections

Gestion des exceptions dans une application

- Gestion des erreurs, les exceptions
- Mécanisme de gestion des exceptions en Java (Exemple d'exceptions Java)
- Gestion locale ou centralisée avec les blocs Try Catch Finally
- Construction des exceptions sur-mesure pour une application

Flux d'entrée/Sortie (I/O)

- Gestion des entrées/sorties
- Les API Java de gestion des flux
- Programmation par exception
- Sérialisation / Désérialisation

Gérer les accès concurrents (Threads)

- Définition d'un thread
- Créer et manipuler les threads
- Les bonnes pratiques

Concepts et syntaxes avancés du langage

- Les types génériques
- Les annotations (méta-data)
- Autoboxing et auto-unboxing des types primitifs
- Méthodes à nombre d'arguments variable

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
- Exploitation des résultats
- Persistance objet, Object Relational Mapping, Hibernate, Toplink

Principes des architectures n-tiers

- Pattern MVC
- Design Patterns:
 - Principe,
 - Singleton,
 - Façade,
 - Observateur,
 - Action, Etat,
 - Factory et IoC,
 - Visiteur.

Détails

Code :

JA-RAD

Durée :

5 jours

Objectifs :

- Etre capable de développer une application en Java
- Savoir installer, configurer et utiliser RAD
- Assimiler les concepts Objet par la pratique de Java
- Savoir gérer les accès concurrents
- Utiliser une base de données avec Java

Public :

- Programmeurs et développeurs débutants
- Directeurs de projets,
- Développeurs (C, C++...)
- Ingénieurs
- Chefs de projets proches du développement.

Pré-requis :

Séminaire « approche orientée objet » ou connaissances équivalentes.

Tarif : 1990 € H.T. par personne (déjeuner inclus)