

Smalltalk pour les développeurs Java

La technologie Smalltalk est très proche de latechnologie Java. En utilisant les similitudes entre les deuxlangages, on peut accélérer l'apprentissage de Smalltalk et allerplus loin dans les détails. Ce cours est conçu spécifiquement pourles développeurs qui maîtrisent Java et qui veulent apprendreefficacement, en profondeur, le langage Smalltalk et êtrerapidement opérationnel.

Pour cela, les concepts sont systématiquement amenés à partir des concepts Java et présentés à l'aide de traductions d'exemples de code Java.

Programme

Historique, philosophie et concepts

- Machine virtuelle, bytecode
- Objet, méthode, variable, classe, metaclass
- Garbage Collector

Environnement de développement

- Console, browser, inspecteur, debugger
- Refactoring, JUnit, CheckStyle, Logging
- Outils spécifiques
- Partage de code, intégration

Syntaxe

- Commentaire
- Type primitif, type de base, littéral
- Envoi de message, opérateur, précédence, enchaînement
- Affectation, retour de méthode
- Définition de classe, attribut, méthode, variable
- Constructeur, initialisation
- Héritage, interface, classe et méthode abstraite
- Pseudo variable this et super
- Tableau
- Espace de nommage, package, visibilité, classpath
- Native, final et autres modificateurs
- Bloc d'instructions, classe interne
- Structures de contrôle conditionnelles (if, else, ?:, switch)
- Structures de contrôle itératives (for, while, do, ?)
- Identité et égalité
- Exceptions (try, catch,...)

Librairie de classes de base

- Object
- Date, Time, Timestamp, Calendar
- Number, Character, String
- Collection, Map, Set, Itérateur
- Stream, Gestion de fichier
- Graphisme

Concepts avancés

- Listener
- Introspection, test sur la hiérarchie
- Invocation dynamique de classe, de méthode, proxy
- Javadoc, annotation, assertion
- Thread, synchronisation
- Réseau, (Socket, protocole et technologies du web)
- JNI, JDBC, XML
- Garbage Collector, finalisation

Spécificités Smalltalk

- Mutabilité, become : changement dynamique de classe
- Accès aux sous-classes
- Extension de classe
- Override
- thisContext

Déploiement

- Exécution, main, classpath
- Technique de déploiement, jar
- Paramétrage, propriétés

Détails

Code :

SM-JV1

Durée :

5 jours

Objectifs :

- Passer rapidement de Java à Smalltalk
- Bénéficier de ses acquis Java pour aller en profondeur dans la compréhension de Smalltalk
- Etre opérationnel dans le développement d'applications Smalltalk

Public :

- Architectes
- Chefs de projets
- Consultants, Ingénieurs

Pré-requis :

Maîtrise du langage Java

Tarif : nous consulter