



Plan de Formation JAVA

PROGRAMMATION EN JAVA

Objectifs

L'objectif de ce cours est de donner aux stagiaires les connaissances nécessaires au développement d'applications en langage Java. A l'issue de ce cours les stagiaires doivent être capables de développer des applications basées sur les concepts orientés objets en Java.

Méthodologie

L'enseignement est composé de 35 heures réparties sur 5 jours soit des séances de 7 heures par jour de formation

Pré requis

Des connaissances en algorithmique peuvent être nécessaires.

Public

Débutants ou expérimentés. Ce cours s'adresse à toute personne désireuse de comprendre le fonctionnement du langage Java.

Cours suivant conseillé

Les cours qu'il est conseillé de suivre sont les suivants :

- La modélisation UML
- La programmation avec J2EE
- Programmation avec Java/Swing

Contenu

1. Présentation du Langage Java

- 1.1. Historique
- 1.2. Comparaison avec les langages C/C++
- 1.3. Notions de machine virtuelle
- 1.4. Exemples de programme

2. Structure d'un programme exécutable en Java

- 2.1. Intérêt de la méthode main()
- 2.2. La commande javac pour la compilation
- 2.3. La commande java pour l'exécution des programmes Java

3. Les types primitifs et les références

- 3.1. Les types entiers
- 3.2. Les types réels en simples et doubles précisions
- 3.3. Notion de référence en Java

4. Les opérateurs

- 4.1. Les opérateurs arithmétiques usuels.
- 4.2. Les opérateurs de comparaison
- 4.3. Signification de l'opérateur de modulo
- 4.4. Les opérateurs Logiques
- 4.5. L'opérateur ternaire
- 4.6. L'opérateur d'affectation

5. Les structures de contrôle en Java

- 5.1. La clause if... else ...
- 5.2. La boucle for()
- 5.3. La boucle while()
- 5.4. La boucle do ... while ...
- 5.5. La clause switch() à choix multiples

6. Les tableaux

- 6.1. Tableaux unidimensionnels
- 6.2. Tableaux multidimensionnels

7. Les principales collections en Java

- 7.1. Les classes Vector, ArrayList et HashMap
- 7.2. L'interface Iterator
- 7.3. L'interface Enumeration
- 7.4. La boucle foreach() en Java
- 7.5. Les collections génériques

8. Les classes en Java

- 8.1. Définition d'une classe en Java
- 8.2. Utilisation d'une classe en Java

9. L'accessibilité

- 9.1. Présentation des modificateurs public, protected et private
- 9.2. Notions d'encapsulation des données

10. Héritage et Classes dérivées

- 10.1. Notions de classe mère et de classe fille
- 10.2. Signification de la clause extends

11. Le polymorphisme

- 11.1. Intérêt du polymorphisme
- 11.2. Mise en œuvre

12. Les exceptions et la gestion des erreurs

- 12.1. Les exceptions contrôlées
- 12.2. Les exceptions non contrôlées
- 12.3. Création d'une exception personnalisée
- 12.4. La classe Exception et l'interface Throwable .

13. Les entrées / sorties en Java

- 13.1. Ecriture dans un fichier de caractères sur disque ou dans un flux
- 13.2. Lecture depuis un fichier de caractères ou depuis un flux de caractères
- 13.3. Lecture de l'entrée standard
- 13.4. Lecture d'un fichier binaire et copie dans un autre fichier binaire

14. La sérialisation d'objets en JAVA

- 14.1. Intérêt de la sérialisation d'objets
- 14.2. L'interface Serializable

15. Les JavaBeans

- 15.1. Intérêt des JavaBeans
- 15.2. Les conventions de nommage de Java.
- 15.3. Les getteurs et les setteurs

16. Le clonage d'objets

- 16.1. Intérêt du clonage
- 16.2. L'interface Cloneable
- 16.3. Mise en œuvre du clonage
 - 16.3.1. Clonage partiel
 - 16.3.2. Clonage en profondeur

17. Les classes enveloppes (wrappers)

- 17.1. Intérêt des classes enveloppes
- 17.2. Les classes Short, Integer, Long, Float, Double.
- 17.3. Les méthodes de conversion des classes enveloppes

18. La gestion des dates et des heures

- 18.1. La classe Date
- 18.2. La classe de formatage de dates : SimpleDateFormat
- 18.3. Les codes formats des dates
- 18.4. Les méthodes de comparaison des dates
- 18.5. La classe Calendar

19. Quelques points à retenir

- 19.1. Constantes et variables de classe
- 19.2. Notions d'interface.

- 19.3. Notions de classes abstraites
- 19.4. Notions de package
- 19.5. Le boxing et l'autoboxing

PROGRAMMATION EN JAVA, Niveau 2: (Option de 3 jours)

20. Programmation parallèle en JAVA avec les Threads

- 1. Présentation et intérêt de la programmation parallèle
 - 1.1. Notion de processus
 - 1.2. Notion de Thread
 - 1.3. Implémentation d'un programme parallèle via la classe Thread
 - 1.3. Implémentation d'un programme parallèle via l'interface Runnable
- 2. Gestion de la concurrence avec les Threads
 - 2.1. La méthode wait()
 - 2.2. Les méthodes notify() et notifyAll()
 - 2.3. Les méthodes join() et la gestion de la synchronisation
 - 2.4. Signification du mot clé synchronized
 - 2.5. Mise en place d'un démon en Java
 - 2.6. Mise en oeuvre

21. Programmation des Applets

- 1. Présentation et intérêt des Applets
- 2. Les méthodes d'une Applet
- 3. Implémentation d'une Applet avec les composants de AWT
- 4. Implémentation d'une Applet avec les composants de SWING
- 5. Implémentation d'une Applet comme un composant de Thread
- 6. Insertion d'une Applet dans une page (X)HTML
- 7. Chargement d'image dans une Applet