

Formation Java maîtrise



La formation "Java maîtrise" présente la syntaxe de Java, la programmation orientée objet en Java, la création d'interfaces utilisateur graphiques (IG), les exceptions, les entrées/sorties de fichiers, les threads et les fonctions réseau.

Cette formation de cinq jours adaptée aux développeurs ayant une première approche de programmation objet leur permettra monter en compétence et avec les concepts orientés objet apprendront à développer des applications Java

Objectifs

- Créer des applications Java qui exploitent les aspects orientés objet du langage (encapsulation, héritage et polymorphisme)
- Apprendre la syntaxe et les principales API de base du langage Java
- Mettre en oeuvre des techniques de traitement des erreurs à l'aide des exceptions
- Créer des interfaces utilisateur graphiques (IG) orientées événements
- Implémenter la fonctionnalité d'entrée/sortie (E/S) pour lire et écrire des données et des fichiers de texte et comprendre les flux d'E/S avancés
- Créer un client TCP/IP (Transmission Control Protocol/Internet Protocol) simple qui communique avec un serveur par l'intermédiaire de sockets.
- Créer des programmes multithreads

Public concerné

- Programmeurs et développeurs confirmé
- Développeurs C, C++

Pré requis

- Savoir créer des programmes dans un langage de programmation quelconque ou avoir suivi le cours JAN0 : Principes de base du langage de programmation Java.

Une formation de 5 jours

Caractéristiques
Tarif : 2175 € HT par personne
Numéro de formateur : 11753687675
Nombre d'heures : 35
Référence : JAN1
Contact : Loic LE FUR
Telephone : 01.41.16.83.70
Email : formation@alterway.fr

Description des modules

num	Module
1	Programmation orientée objet
Détails	<ul style="list-style-type: none">- Modélisation : abstraction, encapsulation et packages- Classe, membre, attribut, méthode, constructeur et package- Appeler une méthode sur un objet particulier- Utiliser la documentation en ligne sur les API Java
2	Identificateurs, mots-clés et types
Détails	<ul style="list-style-type: none">- Commentaires en Java. Génération automatique de documentation avec JavaDoc- Différence entre des identificateurs valides et non valides- Les huit types primitifs- Construire un objet avec 'new'
3	Expressions et contrôle de flux
Détails	<ul style="list-style-type: none">- Reconnaître, décrire et utiliser les opérateurs Java- Faire la différence entre les manières légales et illégales d'instancier des types primitif- Identifier les expressions booléennes et leur caractéristiques dans les constructions de contrôle- Utiliser les instructions if, switch, for, while et do et les formes labellées de break et continue comme structures de contrôle de flux dans un programme
4	Conception de classes
Détails	<ul style="list-style-type: none">- Définir héritage, polymorphisme, surcharge, redéfinition et invocation de méthode virtuelle- Utiliser les modificateurs d'accès protégés et les valeurs par défaut (package-friendly)- Les concepts de constructeur et de surcharges des méthodes- Construction et initialisation d'un objet
5	Fonctions de classes avancées
Détails	<ul style="list-style-type: none">- Créer des variables, des méthodes et des initialisateurs statiques- Créer des classes, des méthodes et des variables finales- Créer et utiliser des types énumérés- Utiliser l'instruction static import- Créer des classes et des méthodes abstraites- Créer et utiliser une interface
6	Exceptions et assertions
Détails	<ul style="list-style-type: none">- Principe et utilité des exceptions- Utiliser les instructions try, catch et finally- Les différentes catégories d'exceptions- Identifier les exceptions courantes- Personnaliser les exceptions- Utiliser les assertions- Activer les assertions pendant l'exécution
7	Structure Collections and Generics
Détails	<ul style="list-style-type: none">- Interfaces de base dans la structure Collections
8	Notions de base d'Entrées/Sorties
Détails	<ul style="list-style-type: none">- Écrire un programme qui utilise les arguments de ligne de commande- Examiner les propriétés d'un flux- Sérialiser et désérialiser des objets- Distinguer les flux de lecture et d'écriture et effectuer des sélections appropriées
9	E/S de console et de fichier
Détails	<ul style="list-style-type: none">- Lire les données de la console- Écrire des données sur la console

- Décrire les fichiers et les Entrées/Sorties de fichier

10 Construire des Interfaces Graphiques avec Swing

- Détails**
- Introduction à Swing
 - Séparation des rôles (contenant et contenu) avec le pattern MVC
 - Les blocs de construction d'une Interface Graphique : JButton, JTextField, JPanel, JFrame
 - Gestionnaires de mise en forme : Layout Manager
 - Gestion des événements : listeners

11 Pratique : création d'applications avec Swing

- Détails**
- Construction d'une barre de menu, d'un menu, et de ses éléments de menu
 - Changement de la couleur et la police d'un composant
 - Construction d'une Applet simple Swing dans une page HTML

12 Threads

- Détails**
- Définir un thread
 - Créer des threads séparés dans un programme Java, en contrôlant le code et les données que chaque thread utilise
 - Contrôler l'exécution d'un thread et écrire un code indépendant de la plate-forme avec les threads
 - Décrire les problèmes qui peuvent se présenter quand plusieurs threads partagent des données
 - Utiliser wait et notify pour communiquer entre plusieurs threads
 - Utiliser synchronized pour protéger les données contre leur corruption

13 Accès au SGBDR et manipulation des données avec JDBC et les classes utilitaires

- Détails**
- Présentation des objets du package java.sql et java.util
 - Connexion à un SGBDR avec Connection, Statement et ResultSet
 - Externalisation des paramètres de connexion avec la classe ResourceBundle et fichiers properties
 - Encapsulation des données dans des collections ou maps : Hashtable et Vector
 - Introduction au mapping Objet-Relationnel

14 Mise en réseau

- Détails**
- Développer du code pour définir la connexion réseau
 - Comprendre TCP/IP
 - Utiliser les classes ServerSocket et Socket pour l'implémentation de clients et serveurs TCP/IP

15 Concepts Objet

- Détails**
- Classes, instances et identité
 - Abstraction et encapsulation
 - Héritage et polymorphisme
 - Interfaces et classes abstraites
 - Notations UML élémentaires : diagramme de classes

16 Syntaxe et construction des Objets

- Détails**
- Constructeurs, classes et packages
 - Attributs et accesseurs get/set, visibilité
 - Méthodes statiques, d'instance, abstraites
 - Types primitifs et promotions de types
 - Tableaux et classe String
 - Structures conditionnelles et boucles itératives